

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО**

Факультет інформатики та обчислювальної техніки

(назва факультету, інституту)

Кафедра автоматизованих систем обробки інформації і управління

(назва кафедри)

"На правах рукопису"

УДК 519.854.2

«До захисту допущено»

Завідувач кафедри

О.А.Павлов
(підпис) (ініціали, прізвище)

“ ” 20 18 р.

МАГІСТЕРСЬКА ДИСЕРТАЦІЯ

на здобуття ступеня магістра

за спеціальністю 122 Комп'ютерні науки та інформаційні технології

(код та назва спеціальності)

спеціалізацією Інформаційні управляючі системи та технології

(код та назва спеціалізації)

на тему: ЗАДАЧА СКЛАДАННЯ РОЗКЛАДУ ВИКОНАННЯ РОБІТ

З ВІДНОШЕННЯМ ПЕРЕДУВАННЯ ПАРАЛЕНЬНИМИ ПРИСТРОЯМИ

ЗА КРИТЕРІЄМ МІНІМІЗАЦІЇ ЗАГАЛЬНОГО ЧАСУ ВИКОНАННЯ РОБІТ

Виконала: студентка VI курсу групи ІС-61

(шифр групи)

Дубок Катерина Валентинівна

(прізвище, ім'я, по батькові)

(підпис)

Науковий керівник к.т.н., ст. викл. Сперкач М.О.

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Консультант к.т.н., доц. Жданова О.Г.

(науковий ступінь, вчене звання, прізвище, ініціали)

(підпис)

Рецензент

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Засвідчую, що у цій магістерській дисертації немає
запозичень з праць інших авторів без відповідних
посилань.

Студент

(підпис)

Київ – 2018

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. ІГОРЯ СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної техніки
(повна назва)

Кафедра автоматизованих систем обробки інформації та управління
(повна назва)

Рівень вищої освіти другий (магістерський)

Спеціальність 122 Комп'ютерні науки та інформаційні технології
(код і назва)

Спеціалізація Інформаційні управляючі системи та технології
(код і назва)

ЗАТВЕРДЖУЮ

В.о. завідувача кафедри

І.П. Муха
(підпис) (ініціали, прізвище)

« 12 » лютого 2018 р.

**ЗАВДАННЯ
на магістерську дисертацію студенту**
Дубок Катерині Валентинівні
(прізвище, ім'я, по батькові)

1. Тема дисертації ЗАДАЧА СКЛАДАННЯ РОЗКЛАДУ ВИКОНАННЯ РОБІТ
З ВІДНОШЕННЯМ ПЕРЕДУВАННЯ ПАРАЛельНИМИ ПРИСТРОЯМИ
ЗА КРИТЕРІЄМ МІНІМІЗАЦІЇ ЗАГАЛЬНОГО ЧАСУ ВИКОНАННЯ РОБІТ

науковий керівник дисертації Сперкач М.О., к.т.н.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від “ 12 ” березня 2018 р. № 885-с

2. Строк подання студентом дисертації “ 5 ” травня 2018 р.

3. Об'єкт дослідження процес календарного планування виконання робіт.

4. Предмет дослідження методи календарного планування виконання робіт
паралельними пристроями.

5. Перелік завдань, які потрібно розробити розробити математичну модель
для класу задач календарного планування, реалізувати наближений алгоритм
розв'язання даної задачі, дослідити ефективність реалізованого алгоритму
шляхом проведення обчислювального експерименту.

6. Орієнтовний перелік ілюстративного матеріалу математична модель,
схема ПДС алгоритму, схеми алгоритмів початкового розкладу A1-A4, ілюстрація
перестановок різних типів, результати експериментів алгоритмів A1-A4,
результати експериментів ПДС алгоритму, архітектура програмного
забезпечення, екранні форми.

7. Орієнтовний перелік публікацій наукових конференціях	одна стаття та двоє тез доповідей на
--	--------------------------------------

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1			
2			
3			
4			

9. Дата видачі завдання “ 12 ” лютого 20 18 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів магістерської дисертації	Примітка
1	Систематизація результатів огляду літератури	15.02.2018	
2	Порівняльний аналіз існуючих методів розв'язання задачі	25.02.2018	
3	Постановка та формалізація математичної моделі задачі	10.03.2018	
4	Модифікація існуючих методів розв'язання задачі	15.03.2018	
5	Розробка інформаційного та програмного забезпечення	01.04.2018	
7	Проведення експериментальних досліджень розроблених алгоритмів	16.04.2018	
8	Оформлення документації	22.04.2018	
9	Подання роботи на попередній захист	23.04.2018	
10	Подання роботи на основний захист	05.05.2018	

Студент

(підпис)

К.В. Дубок

(ініціали, прізвище)

Науковий керівник

(підпис)

М.О. Сперкач

(ініціали, прізвище)

РЕФЕРАТ

Магістерська дисертація: 96 с., 26 рис., 5 табл., 4 додатки, 117 джерел.

Актуальність. Успіх виробництва та його прибутковість на пряму залежить від процесу планування роботи на ньому. Ключовою складовою процесу планування є розробка ефективного плану виконання робіт виробничої діяльності. Для створення таких планів застосовують різні методи, зокрема методи теорії розкладів.

Більшість задач теорії розкладів відносяться до класу NP-повних задач. Виникає проблема в тому, що відомі методи не завжди дозволяють отримувати близькі до оптимальних розклади робіт на виробництві за прийнятний час.

На ряду з іншими NP-повними задачами теорії розкладу стоїть задача складання розкладу виконання робіт з відношенням передування паралельними пристроями з метою мінімізації загального часу виконання робіт. У зв'язку з цим, актуальною є розробка алгоритмів для даної задачі, що забезпечать високу якість отримуваних результатів і не будуть потребувати значних обчислювальних ресурсів.

Мета дослідження – підвищення ефективності календарного планування за рахунок побудови оптимального чи близького до оптимального розкладу виконання робіт з відношенням передування паралельними пристроями шляхом мінімізації загального часу виконання робіт.

Для досягнення мети необхідно виконати наступні **задачі**:

- виконати огляд відомих результатів з розв'язання поставленої в рамках роботи задачі;
- розробити алгоритми побудови початкового розкладу та провести експериментальні дослідження їх ефективності;
- сформулювати достатні умови оптимальності для розв'язуваної задачі;
- розробити алгоритм створення календарного плану мінімізації загального часу виконання робіт з відношенням передування паралельними пристроями;
- розробити програмну реалізацію розробленого алгоритму;
- виконати аналіз отриманих результатів.

Об'єкт дослідження – процес календарного планування виконання робіт.

Предмет дослідження – методи календарного планування виконання робіт паралельними пристроями.

Зв'язок роботи з науковими програмами, планами, темами. Робота виконувалась на кафедрі автоматизованих систем обробки інформації та управління Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського» в рамках теми «Ефективні методи розв'язання задач теорії розкладів» (№ ДР 0117U000919).

Наукова новизна отриманих результатів

Розроблено підходи та методи розв'язання поставленої задачі. Використання створених моделей на основі розроблених достатніх умов оптимальності (ДОУ), дозволяє істотно підвищити швидкодію складання розкладів на підприємстві.

Публікації. Матеріали роботи опубліковані в збірнику праць Міжнародної наукової конференції на тему «Глобальне конкурентне середовище: розвиток сучасних соціально-економічних систем», м. Кишенів, Республіка Молдова, 21 квітня 2017 року [1]; у збірці науково-практичній конференції «Інформатика та обчислювальна техніка-ІОТ- 2018», м. Київ, НТУУ «КПІ ім. Ігоря Сікорського», 23-24 квітня 2018 року [2].

ПАРАЛЕЛЬНІ ПРИСТРОЇ, СКЛАДАННЯ РОЗКЛАДІВ, КАЛЕНДАРНЕ ПЛАНУВАННЯ, ВІДНОШЕННЯ ПЕРЕДУВАННЯ, МІНІМІЗАЦІЯ ЗАГАЛЬНОГО ЧАСУ ВИКОНАННЯ РОБІТ

ABSTRACT

Master's dissertation consists of 96 pages., 26 images., 5 tables., 4 appendixes, 117 referring sources.

Actuality. The success of production and its profitability directly depends on the process of planning work on it. A key component of the planning process is the development of an effective plan for the implementation of job activities. To create such plans, different methods are used. Among them are methods of the theory of schedules.

Majority of scheduling problem belong to the class of NP-complete problems. The problem is that known methods do not always result in solutions with close to the optimal job schedules for an acceptable amount of time.

Along with other NP-complete scheduling problem, the scheduling problem jobs with precedence relation on parallel machines to minimize the makespan. According to this statement, it is important to develop algorithms for this problem, which will ensure the high quality of the results obtained and will not require significant computing resources.

Purpose of the study – to increase scheduling effectiveness by constructing an optimal or near-optimal job schedule with precedence relation on parallel machines to minimize the makespan.

To achieve the goal, it is required to accomplish the following **tasks**:

- provide an overview of the already known solutions for the problem set within the work objective;
- develop initial schedule algorithms and to conduct experimental studies of their effectiveness;
- formulate sufficient optimal conditions for the target problem;
- develop a schedule plan creation algorithm to minimize the makespan with precedence relation on parallel machines;
- develop a software for a schedule plan creation algorithm to minimize the makespan with precedence relation on parallel machines;
- provide result analyzes.

The object of research - the process of job scheduling execution.

Subject of research - methods of calendar planning of the job execution on parallel machines.

Relationship of work with scientific programs, plans, themes. The work was carried out at the Department of Computer-Aided Management And Data Processing Systems of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute" within the objective of the work "Effective Methods for Solving the Problems of the Theory of Schedules" (№ ДП 0117U000919).

Scientific novelty of the obtained results

Approaches and methods of solving the defined task were developed. The use of created models based on developed sufficient conditions of optimality (SCO), can significantly improve the speed of scheduling in the enterprise.

Publications. The materials of the work are published in the collection of articles of the International scientific conference "Global Competitive Environment: Development of Modern Socio-Economic Systems", Kishinev, Republic of Moldova, April 21, 2017 [1]; in the collection of articles of the International scientific-practical conference "Informatics and Computing Technology-IOT-2018", Kyiv, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute" April 23-24, 2018 [2].

PARALLEL MACHINES, SCHEDULING, CALENDAR PLANNING,
RELATION OF PRECEDENCE, MINIMIZE MAKESPAN IN TASKS SCHEDULING

ЗМІСТ

ВСТУП.....	9
1 ОГЛЯД СТАНУ ПРОБЛЕМИ ПЛАНУВАННЯ РОБОТИ ПІДПРИЄМСТВА.....	11
1.1 Процес планування виробництва	11
1.2 Сучасні методології управління виробництвом.....	13
1.3 Оперативне управління основним виробництвом	17
1.4 Системи управління виробництвом	20
Висновок до розділу 1	27
2 ІСНУЮЧІ МЕТОДИ РОЗВ'ЯЗАННЯ ЗАДАЧ ТЕОРІЇ РОЗКЛАДІВ	28
2.1 Огляд методів розв'язання задач теорії розкладів	28
2.1.1 Методи вирішення задач теорії розкладів	28
2.1.2 Огляд робіт запропонованих методів розв'язку	34
2.2 Змістовна постановка задачі дослідження.....	38
Висновок до розділу 2	42
3 РОЗВ'ЯЗАННЯ ЗАДАЧІ СКЛАДАННЯ РОЗКЛАДУ ВИКОНАННЯ РОБІТ З ВІДНОШЕННЯМ ПЕРЕДУВАННЯ ПАРАЛЕЛЬНИМИ ПРИСТРОЯМИ ЗА КРИТЕРІЄМ МІНІМІЗАЦІЇ ЗАГАЛЬНОГО ЧАСУ ВИКОНАННЯ РОБІТ	43
3.1 Задача складання розкладу виконання робіт з відношенням передування паралельними пристроями різної продуктивності за критерієм мінімізації загального часу виконання робіт	44
3.1.1 Математична постановка задачі	44
3.1.2 Дослідження властивостей задачі	45
3.1.2.1 Достатні умови оптимальності розкладів	48
3.1.3 Процес розв'язання вихідної задачі	49
3.1.3.1 Знаходження точки поділу палети.....	49
3.1.3.2 Побудова розкладу виконання робіт	52
3.1.3.2.1 Побудова початкового розкладу	52
3.1.3.2.2 Розробка множини операцій обміну	55
3.1.3.2.3 ПДС-алгоритм розв'язання задачі	63
3.2 Опис експериментальних досліджень ефективності алгоритмів	64
3.2.1 Класифікація задач	64
3.2.2 Результати експериментів	65
3.2.2.1 Тривалість робіт	66
3.2.2.2 Кількість робіт.....	67
3.2.2.3 Кількість ланцюгів	69
Висновок до розділу 3	70
4 ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ СИСТЕМИ.....	72
4.1 Реалізація програмного продукту	74
4.1.1 Опис пакетів та класів програмного продукту.....	74
4.1.2 Опис бази даних.....	75
4.2 Обґрунтування вибору технологій розробки	75
4.3 Огляд програмної реалізації продукту.....	77
Висновок до розділу 4	80
ВИСНОВКИ	81
ПЕРЕЛІК ПОСИЛАНЬ.....	83
ДОДАТКИ.....	95

ДОДАТОК А Ілюстрація перестановок різних типів між паралельними пристроями для мінімізації максимального з виступів	95
ДОДАТОК Б Приклад розв’язання задачі складання розкладу виконання робі з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт	97
ДОДАТОК В Результати проведення експериментів.....	105
ДОДАТОК Г Графічний матеріал	108
ПЛАКАТ 1 Математична модель	109
ПЛАКАТ 2 Схема ПДС алгоритму	110
ПЛАКАТ 3 Схеми алгоритмів початкового розкладу А1-А4.....	111
ПЛАКАТ 4 Ілюстрація перестановок різних типів.....	112
ПЛАКАТ 5 Результати експериментів алгоритмів А1-А4.....	113
ПЛАКАТ 6 Результати експериментів ПДС алгоритму.....	114
ПЛАКАТ 7 Архітектура програмного забезпечення.....	115
ПЛАКАТ 8 Екранні форми	116

ВСТУП

Актуальність. Потреба в пошуку нових підходів до ефективного функціонування виробництва виникає за рахунок стрімкого економічного розвитку. Успіх виробництва та його прибутковість на пряму залежить від процесу планування роботи на ньому. Ключовою складовою процесу планування є розробка ефективного плану виконання робіт виробничої діяльності. Для створення таких планів застосовують різні методи, зокрема методи теорії розкладів.

Більшість задач теорії розкладів відносяться до класу NP-повних задач, тобто таких, що час на їх розв'язання в найгіршому випадку буде експоненційно зростати. Тому практичні завдання вирішуються наближеними або евристичними методами.

На сьогодні зусиллями багатьох науковців як В.М. Глушкова, В.С. Михалевича, М.З. Згуровського, І.В. Сергієнка, Н.З. Шора, В.С. Танаєва, В.В. Шкурби, В.Л. Волковича, О.А. Павлова, Р.Л. Грема були розроблені базові алгоритми для розв'язання основних класів задач оперативно-календарного планування. Але відомі методи не завжди дозволяють отримувати близькі до оптимальних рішення за прийнятний час, до того ж виникає потреба в інструментах формування і перевірки умов розв'язання все нових класів задач.

На ряду з іншими NP-повними задачами теорії розкладу стоїть задача складання розкладу виконання робіт з відношенням передування паралельними пристроями з метою мінімізації загального часу виконання робіт. У зв'язку з цим, актуальною є розробка алгоритмів для даної задачі, що забезпечать високу якість отримуваних результатів і не будуть потребувати значних обчислювальних ресурсів.

Зв'язок роботи з науковими програмами, планами, темами. Робота виконувалась на кафедрі автоматизованих систем обробки інформації та управління Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського» в рамках теми «Ефективні методи розв'язання задач теорії розкладів» (№ ДР 0117U000919).

Мета дослідження – підвищення ефективності календарного планування за рахунок побудови оптимального чи близького до оптимального розкладу виконання

робіт з відношенням передування паралельними пристроями шляхом мінімізації загального часу виконання робіт.

Для досягнення мети необхідно виконати наступні **задачі**:

- виконати огляд відомих результатів з розв’язання поставленої в рамках роботи задачі;
- розробити алгоритми побудови початкового розкладу та провести експериментальні дослідження їх ефективності;
- сформулювати достатні умови оптимальності для розв’язуваної задачі;
- розробити алгоритм створення календарного плану мінімізації загального часу виконання робіт з відношенням передування паралельними пристроями;
- розробити програмну реалізацію розробленого алгоритму;
- виконати аналіз отриманих результатів.

Наукова новизна отриманих результатів

Розроблено підходи та методи розв’язання поставленої задачі. Використання створених моделей на основі розроблених достатніх умов оптимальності (ДОУ) дозволяє істотно підвищити швидкодію складання розкладів на підприємстві.

1 ОГЛЯД СТАНУ ПРОБЛЕМИ ПЛАНУВАННЯ РОБОТИ ПІДПРИЄМСТВА

На сьогоднішній день однією з основних складових нормальної роботи підприємства та її конкурентоспроможності серед невпинно зростаючої кількості підприємств та швидкої зміни технологічних процесів є оптимальне планування виробничого процесу. За сучасних умов підприємство повинно швидко реагувати на потреби ринку та бути адаптованим до різких змін у виробництві. Це все обумовлює необхідність впровадження на підприємствах автоматизованих систем управління виробництвом, що в свою чергу сприяє покращенню організації на виробництві, узгодженості роботи усіх його відділів.

Однією з таких систем є автоматизована система планування завантаження потужностей виробництва. У даній роботі розглядається задача заповнення плоскої конструкції за допомогою пристроїв за мінімальний час. Ця задача вирішується механізмами розділу дискретної математики – теорії розкладів, що займається проблемами планування. В термінах теорії розкладів, проблема має наступне формулювання: задача складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт. Дана задача є NP-повною, тобто для неї не знайдено рішення за поліноміальний час. Для її розв'язання запропоновано використати ПДС-методологію, що дозволить знайти рішення та оцінити його оптимальність за прийнятний час.

Розглянемо далі процес планування виробництва.

1.1 Процес планування виробництва

Успіх розвитку та функціонування кожної компанії залежить від багатьох складових, основною серед яких є ефективне планування її роботи.

Процес планування знаходить своє логічне завершення в плані. Планом називається офіційний документ, у якому відображаються прогнози розвитку підприємства і окремих сторін його діяльності у майбутньому; проміжні і кінцеві

задачі та цілі, що стоять перед ним і його окремими підрозділами у відповідній сфері; механізми координації поточної діяльності і розподілу ресурсів; стратегії на випадок надзвичайних обставин [3].

Можна видалити три основні типи планів:

- плани-цілі;
- плани для повторювальних дій;
- плани для неповторювальних дій.

Відповідно виконання сформованого плану призведе до підвищення продуктивності компанії та зменшення її витрат.

При побудові розкладу для виробництва на підприємстві виникає проблема планування. Проблемами планування займається розділ дискретної математики – теорія розкладів.

Оскільки, в оптимальному розкладі (плані) на кожному пристрої може бути свій порядок обробки, то з ростом розмірності задачі величина варіантів розташування робіт зростає настільки швидко, що отримати рішення навіть з використанням найбільш потужних електронних обчислювальних машин (ЕОМ) можна лише для задач невеликих розмірів. Ця обставина викликала багато спроб застосувати будь-який інший метод вирішення задачі, причому природною оцінкою будь-якого методу служить трудомісткість рішення.

Існують різні варіанти задач теорії розкладів, частина з них є NP-повними, частина належить до класу поліноміальних задач, для частини завдань так і не вдалося довести приналежність до певного класу складності. Існує гіпотеза, що задача, яка допускає переривання, не складніша за задачі без переривань. Для більшості задач гіпотеза залишається правдивою, крім однієї, де для варіанту без переривання доведена його приналежність до класу поліноміальних задач, в той час як для аналогічної задачі з перериваннями не існує доказів приналежності до якого-небудь класу складності.

Дана задача складання розкладу виконання робіт з відношенням передування паралельними пристроями з метою мінімізації загального часу виконання робіт є NP-повною задачею без переривань.

Планування виробництва можна розділити на наступні категорії [4]:

- визначення черговості виконання завдань на одному пристрої (single-machine-problem або one-machine sequencing problem);
- визначення черговості виконання завдань на паралельних пристроях (single machine scheduling problem with parallel machines);
- визначення черговості виконання множини завдань для групи розміщених один за одним пристроїв, тобто конвеєр (flow-shop- scheduling);
- визначення черговості виконання множини завдань для групи розміщених один за одним пристроїв, коли кожне завдання, яке повинно на них виконуватися, складається із множини операцій та виконання кожної із цих операцій кожного із завдань можливе тільки на одному строго фіксованому пристрої (job shop scheduling);
- визначення черговості виконання множини завдань, де кожне завдання складається з множини операцій і на цій множині немає умови передування (open shop scheduling).

1.2 Сучасні методології управління виробництвом

Ефективність управління сучасним підприємством та його результат значною мірою залежить від того, які методології (методи) управління виробництвом застосовуються.

За допомогою методології управління керуюча система регулює, об'єднує та координує поведінку та дії всіх підпорядкованих суб'єктів, також забезпечує вибір адекватних методів управління тощо.

Принципи управління – це правила, норми управлінської діяльності, відповідно до яких створюється, функціонує та розвивається система менеджменту організації. Принципи управління є узагальненням практичного позитивного управлінського

досвіду і ґрунтуються на певних законах і закономірностях розвитку методології управління виробництвом [5].

Наведемо огляд існуючих методологій управління виробництвом.

Домінуюча методологія управління виробництвом минулого століття має назву «Фордизм». Фордизм – це модель масового виробництва стандартизованих товарів на складальних конвеєрах з використанням низько кваліфікованих працівників, зайнятих простими операціями. Один з основних постулатів фордизму: «Виробляти великі партії виробів вигідніше, ніж дрібні», міцно укорінився серед управлінців XX століття

З другої половини минулого століття було багато спроб модифікувати фордизькі моделі. Усі рішення та відкриття з часом склалися в нову методологію, що отримала назву ошадне виробництво (Lean Manufacturing (LM)).

У кінці минулого століття з'явилася велика кількість різних методологій та парадигм з управління виробництва, серед яких докладніше розглянемо наступні: Швидкореагуюче виробництво (Quick Response Manufacturing (QRM)) та Активне виробництво (Agile Manufacturing (AM)). Сучасні, більш гнучкі в порівнянні з фордизмом методології, часто об'єднують терміном «постфордизм».

Ошадне виробництво (Lean Manufacturing, LM)

Метод застосовується для оптимізації виробництва і підвищення конкурентоспроможності. Метою методу є побудова виробництва, здатного швидко відповідати на вимоги споживачів і отримувати прибуток при будь-якій зміні ринку, у тому числі при падінні попиту. Створення досконалої виробничої системи, яка б при надходженні замовлення миттєво поставляла необхідну продукцію, і при цьому не відбувалося накопичення проміжних запасів.

Дана методологія спрямована на боротьбу з втратами у всіх їх проявах: зайві складські запаси, час простою, зайві переміщення, враховуючи при цьому зручність і безпеку виконання операцій для персоналу. Сьогодні ця система відома як виробнича система Toyota (Toyota Production System – TPS), принципи та інструменти якої знайшли відображення в її американському варіанті – системі ошадного виробництва

(Lean Production). Багато з її складових, були ще й і в радянському варіанті – наукової організації праці (НОП).

Згодом, у рамках концепції ощадного виробництва було виділено безліч напрямків, кожен з яких представляє собою окремий метод: Потік одиничних виробів, Канбан, Всезагальне обслуговування обладнання (Total Productive Management (TPM)), Система 5S, Швидке переналагодження (SMED), Кайдзен, Захист від дурня, Управління персоналом, Точно в строк (Just In Time (JIT)).

Деякі методи самі претендують на статус самостійної виробничої методології або концепції, тому розглянемо деякі популярні з них, а саме концепцію «точно в строк».

Сенс роботи за концепцією «точно в строк» полягає в тому, щоб на всіх фазах виробничого циклу необхідний напівфабрикат до місця подальшої виробничої операції надходив саме тоді, коли це потрібно. Така концепція є в якійсь мірі «витягаючою», тобто такою, коли ділянки, розташовані на наступних етапах виробничого циклу, як би витягають необхідну їм продукцію з попередніх ділянок.

Головна мета виробничої системи «точно в строк» – забезпечити гнучку перебудову виробництва при зміні попиту. Така система забезпечує оперативне регулювання кількості виробленої продукції на кожній стадії виробництва.

Відомі різні модифікації системи «точно в строк», що застосовуються у світовій практиці. В їх основі лежить розробка, вперше застосована японською компанією «Тойота» і отримала широку популярність як система «Канбан».

Швидкореагуюче виробництво (Quick Response Manufacturing, QRM)

Швидке зростання великої кількості продукції, що пропонують виробники своїм замовникам, пов'язано зокрема з: розвитком CAD / CAM (системи автоматизованого проектування і виробництва), що дозволяє компаніям розробляти «під клієнта», а потім виробляти продукцію без несення високих додаткових витрат; розвитком Інтернет, який дозволяє замовнику без особливих

Отже, швидко реагуюче виробництво (QRM) – стратегія, що використовується компаніями для скорочення часу виконання замовлення та яка охоплює все

підприємство. Мета QRM – скоротити час виконання замовлення за рахунок усіх операцій компанії, як внутрішніх, так і зовнішніх.

Якщо ощадне виробництво та інші схожі методології, засновані на зниженні витрат, спрямовані на скорочення реального часу роботи, то QRM орієнтований на зниження всього часу виконання замовлення.

Основні концепції QRM полягають у тому, що бізнес побудований на основі виконання роботи «для складу» (коли, щоб швидше виконати замовлення, основна номенклатура продуктів виготовляється заздалегідь та кладеться на склад), що через помилки планування та мінливості попиту призводить до збільшення КШВ, і в результаті це приводить до того, що компанія не може швидко реагувати на потреби клієнтів. Якщо сильно спростити – краще інвестувати у верстати і стандартно швидко реалізацію замовлень, ніж у склади. Тому був започаткований перехід від функціональних цехів до QRM-осередків. У філософії QRM-осередків можна простежити деяку аналогію з Scrum командою. Орієнтація працівників усіх підрозділів на єдину мету – зниження часових витрат, а звідси і єдині параметри оцінювання роботи для її досягнення, згуртувати команду працівників.

Активне (гнучке) виробництво (Agile Manufacturing, AM)

AM – це стратегія управління компанією, мета якої зробити виробничу компанію більш стійкою до криз, змін попиту та інших непрогнозованих змін.

Для компаній, що працюють по AM, властива здатність швидко реконфігурувати трудові та матеріальні ресурси, щоб не втратити можливості заробити та уникнути неприємностей. Основною перевагою концепції AM є вміння оперативно підлаштовуватися під ситуацію, що змінюється і працювати в умовах невизначеності на ринку. AM підходить для галузей, де високий рівень невизначеності (наприклад IT, споживча електроніка та ін.).

Відбувається постійна готовність до змін і відповідь на них за допомогою сценарних стратегій. Сценарії: якнайбільше інтелектуальних ресурсів і якомога менше матеріальних; постійна група кросфункціональних фахівців; група фахівців на проектах на договірній основі, а також винесення непрофільних робіт на аутсорс;

штат працівників не роздувається, а люди становлять ядро компанії та зацікавлені у справі.

Таким чином, особливу увагу, в компаніях типу АМ приділяється мінімізації втрат від можливих, несподіваних негативних змін, таких як втрата контрактів або ринку збуту вироблених виробів. Одночасно, дисциплінарна, швидко розширювана команда й розгалужена партнерська мережа створюють передумови для того, щоб швидко реагувати на несподівані зміни.

Проведемо порівняння наведених методологій. Три розглянуті методології відрізняються в першу чергу стратегічною орієнтацією. LM націлене на створення більшого за допомогою мінімальних засобів. Іншими словами, LM постійно виявляє втрати будь-якого плану та їх ліквідує. QRM націлене на єдину мету – зменшення часу циклу виробництва від отримання заявки і до здачі продукту замовнику. У АМ головною метою є вдосконалення можливостей для роботи в умовах невизначеності та мінливості ринку.

З урахуванням особливостей дрібносерійного типу виробництва та поставленої мети даної роботи розробка інформаційної технології оперативно-календарного планування дрібносерійного виробництва буде виконуватися за концепцією «точно в строк», яка відноситься до методології LM та може забезпечити гнучку перебудову виробництва при зміні попиту, також вона забезпечує оперативне регулювання виготовлення продукції на кожній стадії виробництва, що призводить до зменшення обсягів витрат на виробництво та збільшення прибутку.

1.3 Оперативне управління основним виробництвом

Планування роботи та управління виробництвом підрозділяється на техніко-економічне та оперативне планування. Оперативне планування підрозділяється на оперативно-виробниче планування та оперативне планування допоміжного виробництва. До складу оперативно-виробничого планування входить оперативно-календарне планування, диспетчеризація [6].

Завданням оперативного планування є організація взаємоузгодженої роботи всіх виробничих підрозділів підприємства для забезпечення своєчасного виконання

планового завдання при економному використанні ресурсів та високій якості продукції, тобто досягнення найкращих кінцевих результатів виробництва.

Змістом оперативного управління основним виробництвом є: встановлення місця та часу виготовлення виробів, деталей; облік фактичного ходу виробничого процесу; визначення відхилень від заздалегідь встановленого плану; регулювання ходу виробництва.

Система оперативного управління основним виробництвом (ОУ ОВ) являє собою складну організаційно-планову систему, що включає: функціональну, елементну та організаційну підсистеми.

Наукова обґрунтованість системи ОУ ОВ припускає визначення: вибору елементів системи; вибору та розрахунку календарно-планових нормативів; побудови об'ємних і оперативно-календарних планів; системи контролю та регулювання виробництва; достовірність вихідних даних.

Для оперативного планування виробництва застосовують системи планування і контролю виробництва, що сприяють організації рівномірної, ритмічної взаємоузгодженої роботи всіх виробничих підрозділів підприємства. Їх формування та застосування описані у роботах [7-15].

Система планування і контролю виробництва складається з п'яти основних рівнів: розробка стратегічного бізнес-плану, формування плану виробництва, створення головного календарного плану виробництва, розробка плану потреби в ресурсах, формування переліку закупівель, здійснення планів та контроль над виробничою діяльністю.

Метою ОУ ОВ є забезпечення суворого виконання заданого плану випуску продукції за кількістю та номенклатурою і у встановлені строки на основі оптимального використання виробничих ресурсів, а також шляхом виявлення та мобілізації внутрішньовиробничих резервів.

Для реалізації цієї мети необхідна науково-обґрунтована побудована та функціональна система ОУ ОВ, яка має забезпечити вирішення наступних завдань:

- повне, комплектне і рівномірне виконання виробничої програми при дотриманні директивних строків випуску продукції;

- повне та найбільш оптимальне використання засобів виробництва і трудових ресурсів; максимальне прискорення виробництва та забезпечення максимального використання оборотних коштів у стадії виробництва;
- забезпечення умов, що сприяють розвитку передових форм організації праці у виробництві;
- автоматизація виконання основних планово-облікових, облікових робіт та отримання документації.

До функцій системи ОУ ОВ відносяться: управління як процес прийняття рішень; планування як процес визначення лінії поведінки об'єкта управління; облік як процес контролю, аналізу та виявлення відхилень від заданої планом лінії поведінки об'єкта; регулювання як процес локалізації виникаючих відхилень і збереження заданої лінії поведінки керованого об'єкта [16].

Система ОУ ОВ включає: об'ємне планування; оперативно-календарне планування; оперативний облік; поточний контроль і регулювання.

У процесі об'ємного планування проводиться розподіл річної виробничої програми виробництва в об'ємному і натуральному вираженні між цехами та дільницями відповідно до виділених їм трудових та матеріальних ресурсів. Для вирішення завдань об'ємного планування використовуються методи математичного програмування.

Оперативно-календарне планування – продовження та розвиток об'ємного планування. На цьому етапі об'єктом планування є окремі вироби, складальні одиниці, деталі, деталеоперації. Воно ґрунтується на певних нормативах (розміри партій деталей, складальних одиниць та виробів; тривалість виробничих циклів, випередження запуску і випуску партій деталей; нормативи незавершеного виробництва), що дозволяють здійснити зв'язок календарних планів та узгодження роботи взаємопов'язаних робочих місць, дільниць, цехів.

Оперативно-календарне планування здійснюється як на заводському, так і на внутрішньоцеховому рівнях. У завдання заводського оперативно-календарного планування входить визначення кількості та часу передачі по планованих позиціях

(деталей, виробів) з цеху в цех по місяцях або кварталах.

До завдань внутрішньоцехового оперативного обліку відносяться: визначення виконання плану випуску деталей і складальних одиниць, а також плану виготовлення деталеоперацій і складальних одиниць дільницями; облік наявності та надходження матеріалу, незавершеного виробництва на дільниці, цеха; облік браку та використання обладнання на дільницях; визначення виконання змінно-добових завдань дільницями і т.п.

Поточний контроль та регулювання виробництва відносяться до виробничої диспетчеризації, яка є складовою частиною оперативного управління. Вона включає: безперервний облік фактичного ходу робіт з виконання встановленого плану виробництва; прийняття оперативних заходів щодо попередження та усунення відхилень від плану; виявлення та аналіз причин відхилень від встановлених календарних графіків виробництва; координацію поточної роботи; організаційне управління оперативної підготовки всього необхідного для виконання календарних графіків виробництва.

Ефективність виконання завдань з організації та управління виробництвом багато в чому визначається забезпеченням його технічними засобами, які повинні забезпечувати збір та обробку інформації з оперативного обліку, контролю та регулювання ходу виробництва і передачу її за допомогою засобів зв'язку у відповідні підрозділи підприємства [5].

1.4 Системи управління виробництвом

Вже давно більшість компаній шукають способи спростити управління виробничими процесами. Для цього використовують різні системи управління виробництвом.

Першим кроком в розробці таких систем стало створення концепції MRP (Materials Resource Planning), яка підтримувала планування матеріальних ресурсів для виробництва. Основна мета концепції MRP полягала в мінімізації витрат, пов'язаних зі складськими запасами (в тому числі і на різних дільницях виробництва). В основі цієї концепції лежить поняття BOM (Bill Of Material – специфікація виробів), яка

відображає залежність попиту на сировину, напівфабрикати та інші продукти від плану випуску готової продукції. При цьому дуже важливу роль відіграє час, для обліку якого необхідно мати чітке уявлення про технологічний ланцюжок випуску продукції, тобто знати, яка послідовність і тривалість операцій. На підставі плану випуску продукції, BOM і технологічного ланцюжка здійснюється розрахунок потреби в матеріалах до конкретних строків.

Однак у концепції MRP є серйозний недолік, а саме те, що при розрахунку в рамках цієї концепції не враховуються потреби в матеріалах, ні наявні виробничі потужності, ні їх завантаження, ні вартість робочої сили. Цей недолік був виправлений в концепції MRP II (Manufacturing Resource Planning – планування виробничих ресурсів). Поняття «MRP II» використовується для позначення відмінності «плану виробничих ресурсів» (MRP II) від «плану потреби в ресурсах» (MRP). MRP II дозволяла враховувати і планувати всі виробничі ресурси підприємства – сировину, матеріали, обладнання, персонал і т.д. MRP II забезпечує координацію маркетингу і виробництва.

Маркетинговий, фінансовий і виробничий відділ погоджують загальний, придатний для роботи план, виражений у виробничому плані.

Виробничий план встановлює загальні рівні виробництва і матеріально-виробничих запасів на відповідний горизонт планування строків. Першочергова мета полягає в тому, щоб визначити норми виробництва, які дозволять виконати поставлені в стратегічному бізнес-плані завдання.

Відділи маркетингу і виробництва повинні щотижня і щодня взаємодіяти з метою коригування плану з урахуванням змін (зміна розміру замовлення, скасування замовлення або затвердження відповідну дату поставки), що відбуваються. Зміни такого роду здійснюються в рамках генерального календарного плану виробництва [17].

Завдання, які підлягають виконанню при календарному плануванні виробництва, полягають у наступному [18, 19-22]: вибір ресурсів для виконання усіх поставлених завдань, агрегування та організація упорядкування різних видів ресурсів, детальне планування використання ресурсів (пристроїв) у часі, вирішення проблем

упорядкування виконання завдань на кожній із одиниць обладнання, побудова календарних планів роботи обладнання та виконання завдань, календарне планування роботи виробничого транспорту [23].

Оперативно-календарне планування складається з чотирьох функціональних етапів [24]:

- 1) об'ємно-календарне планування (Master Production Schedule, MPS);
- 2) баланс виробничих потужностей (Capacity Planning Problem, CPP);
- 3) розрахунок виробничого розкладу (Production Scheduling, PS);
- 4) групування деталей і складальних одиниць та обладнання (Group Technology, GT).

Менеджери відділів маркетингу та виробництва можуть вносити зміни в генеральні календарні плани виробництва з урахуванням змін прогнозованого попиту. Управління підприємства може змінювати виробничий план відповідно до загальних змін попиту або положення з ресурсами. Проте, всі співробітники працюють в рамках системи MRP II. Вона служить механізмом координації роботи маркетингового, фінансового, виробничого та інших підрозділів компанії. MRP II являє собою метод ефективного планування всіх ресурсів виробничого підприємства [17].

По мірі розвитку концепції MRP II до неї поступово додавалися можливості обліку інших витрат підприємства. Так з'явилася концепція ERP (Enterprise Resource Planning – планування ресурсів підприємства). В основі ERP лежить принцип створення єдиного сховища даних, що містить всю ділову інформацію, накопичену організацією в процесі ведення бізнесу, зокрема фінансову інформацію, дані, пов'язані з виробництвом, управлінням персоналом, і будь-які інші дані, які стають одночасно доступними для всіх працівників, що володіють відповідними повноваженнями.

У дев'ятому виданні «Словника APICS» Американської Асоціації Контролю над Виробництвом та матеріально-виробничих запасів (APICS) дається таке визначення ERP: призначена для ведення звітності інформаційна система

ідентифікації та планування підприємства – глобальних ресурсів, необхідних для виробництва, транспортування і складання звітів за замовленнями клієнтів.

У роботі [25] ERP-система має наступне визначення – це інтегрована система на базі інформаційних технологій для управління внутрішніми і зовнішніми ресурсами підприємства.

Концепція ERP знайшла широке застосування, оскільки планування ресурсів дозволяло скоротити час випуску продукції, знизити рівень товарно-матеріальних запасів, а також поліпшити зворотний зв'язок зі споживачем при одночасному скороченні адміністративного апарату. Стандарт ERP дозволив об'єднати всі ресурси підприємства і підвищити ефективність управління ними.

Для повноцінної експлуатації повинні бути передбачені програми для планування, календарного планування, калькуляції собівартості та інше на всіх рівнях організації, у робочих центрах, відділах, підрозділах та для всіх їх разом. Важливо відзначити, що ERP охоплює компанію цілком, а MRP II відноситься до виробництва [17]. Проектування та застосування ERP-систем широко описані у роботах [26, 27].

На теперішній час практично всі сучасні західні системи управління виробництвом базуються на концепції ERP і відповідають її рекомендаціям. Ці рекомендації виробляються американською громадською організацією APICS, що об'єднує виробників, консультантів у галузі управління виробництвом, а також розробників програмного забезпечення (ПЗ).

Одним із стандартів систем управління підприємствами є CSRP (Customer Synchronized Resource Planning), крім усього іншого, він охоплює і взаємодію з клієнтами: оформлення нарядів / замовлень і технічних завдань, підтримка замовника на місцях і т.п. Таким чином, якщо стандарти MRP, MRP II та ERP орієнтовані на внутрішню організацію підприємства, то стандарт CSRP включає в себе повний цикл – від проектування майбутнього виробу, з урахуванням вимог замовника, до гарантійного і сервісного обслуговування після продажу. Суть концепції CSRP головним чином полягає в тому, щоб інтегрувати замовника в систему управління підприємством. Відповідно до даної концепції не відділ збуту, а безпосередньо сам покупець розміщує замовлення на виготовлення продукції, сам відповідає за

правильність його виконання і при необхідності відстежує дотримання строків виробництва і поставки. При цьому, саме підприємство може дуже чітко відстежувати тенденції попиту на його продукцію.

На сьогодні існує безліч програмних продуктів, які дозволяють вирішувати ті чи інші завдання календарного планування. Наприклад, функціональні етапи календарного планування MPS та CPP виконуються, як правило, системами класу ERP (Enterprise Resource Planning), етапи PS та GT реалізуються засобами MES (Manufacturing Execution Systems). Хоча для CPP і PS етапів календарного планування виробництва також успішно використовують APS (Advanced Planning Systems) [24].

На передових як вітчизняних так і зарубіжних підприємствах широко впроваджуються:

- системи управління ресурсами підприємства (ERP) [24], що формують в автоматизованому режимі номенклатурні плани виробництва;
- програми для вдосконаленого планування (Advanced Planning & Scheduling, APS) [24], що представляє собою концепцію оптимізованого (або синхронного) виробничого планування, головною особливістю якої є можливість швидкого складання планів з урахуванням наявних ресурсів і виробничих обмежень (переналагодження обладнання, доступність оснащення, зв'язок між пристроями та ін.) і швидкого перепланування за заздалегідь складеним сценарієм оптимізації, концепція APS поєднала в собі основні елементи концепцій MRP, MRP II, Finite Capacity Scheduling (FCS);
- виконавчі виробничі системи (Manufacturing Execution System, MES) [24], що пов'язують воедино всі бізнес-процеси підприємства з виробничими процесами, які підвищують ефективність використання верстатної системи;
- системи управління трудовими ресурсами (Human Resource Management, HRM) [28], їх ділять на «розрахункові», «облікові» та системи управління трудовими ресурсами за рівнями автоматизації управління персоналом, які відповідають етапам розвитку прикладних програмних рішень для кадрових служб;

- системи для ефективного управління складом (Warehouse Management System, WMS) [29], що підтримують усі складські бізнес-процеси, якісне планування операцій, розробку раціональної топології складу, управління ресурсами, номенклатурою, впровадження передових методів для обробки і зберігання запасів;
- системи управління ланцюгами поставок (Supply Chain Management, SCM) [30], які призначені для автоматизації та управління всіма етапами постачання ресурсами підприємства і для контролю за рухом товару на підприємстві;
- корпоративна інформаційна система управління відносинами з клієнтами (Customer Relationship Management, CRM) [31], що дає можливість не просто автоматизувати взаємодію з клієнтами і процес продажів, а вибудовувати їх роботу таким чином, щоб отримувати максимальний результат;
- система управління життєвим циклом продукції (Product Lifecycle Management, PLM) [32], являє собою методологію застосування сучасних інформаційних технологій для підвищення конкурентоспроможності промислових підприємств.

Основними компонентами PLM-системи є:

- PDM-система (Product Data Management) – система управління даними про виріб, є основою PLM, призначена для зберігання і управління даними;
- CAD-система (Computer Aided Design) – проектування виробів;
- CAE-система (Computer Aided Engineering) – інженерні розрахунки;
- CAPP-система (Computer Aided Production Planning) – розробка прогресивних технологічних процесів;
- CAM-система (Computer Aided Manufacturing) – розробка керуючих програм для верстатів;
- MPM-система (Manufacturing Process Management) – моделювання та аналіз виробництва виробів;
- система скорочення витрат (Enterprise Asset Management, EAM) [33] орієнтована на скорочення витрат, пов'язаних із обслуговуванням устаткування і підвищенням продуктивності.

Інтеграція різних систем привела до розмивання меж між ними. Наслідком цього процесу стало нечітке розуміння, навіть у середовищі IT-фахівців, функціональної спеціалізації різних систем планування та управління виробництвом. Виробники ERP-систем стали заявляти, що їхні продукти тепер «легко вирішують» всі MES-завдання, а виробники MES-систем, у свою чергу, почали переконувати представників виробництва, що саме їх системи прийшли на зміну «застарілим» системам класу ERP. Продавці APS твердять про те, що, системи класу APS легко можуть замінити як ERP, так і MES разом узяті [24].

Розглянемо існуючі функціональні відмінності між декількома розглянутими системами планування виробництва.

Система ERP виконує об'ємно-календарне планування (MPS), попутно вирішуючи завдання балансу виробничих потужностей (CPP).

Система APS уточнює розраховані в ERP планові завдання, розподіляючи (оптимізуючи) їх по робочих місцях. На цьому етапі можливе внесення корекцій у вихідний MPS. Оскільки послідовність цих завдань, як правило, залежить від залученого в контур планування ланцюжка поставок матеріалів і комплектуючих (SCM), то порушувати цю послідовність всередині цеху вже не можна без ризику зруйнувати основний виробничий план.

Система MES за рахунок можливостей багатокритеріальної оптимізації дозволяє, варіюючи пріоритети різних завдань, вирішити задачу максимізації швидкості їх проходження через робочі місця. При цьому допускається не тільки перевпорядкування завдань, а й перерозподіл деяких з них на інші взаємозамінні робочі місця. Відзначимо, що на MES-рівні ніяких змін у загальний виробничий план вносити не можна [24].

Програмне забезпечення, розробка якого розглядається у даній роботі, дозволяє вирішити задачу складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт.

Висновок до розділу 1

У даному розділі було розглянуто таку основну складову функціонування кожної компанії як: процес планування роботи виробництва та проблему планування, що виникає при побудові плану роботи підприємства. Було описано стан проблеми планування роботи підприємства на сьогоднішній день. Наведено сучасні методології управління виробництвом та оперативне управління основним виробництвом, основною складовою якого є оперативно-календарне планування та диспетчеризація, для яких і розробляється інформаційна технологія. Також було наведено різні системи управління виробництвом, як спосіб спростити управління виробничими процесами.

2 ІСНУЮЧІ МЕТОДИ РОЗВ’ЯЗАННЯ ЗАДАЧ ТЕОРІЇ РОЗКЛАДІВ

На даний час існують різні методи розв’язку задач теорії розкладу. Проведемо огляд запропонованих методів розв’язання.

2.1 Огляд методів розв’язання задач теорії розкладів

2.1.1 Методи вирішення задач теорії розкладів

Велику кількість методів дискретної оптимізації можна розділити на дві групи: точні та наближені. На даний момент цей поділ є умовним, адже багато точних методів можуть застосовуватися як наближені, а наближені методи при певних умовах можуть застосовуватися як точні або їх складова частина [16]. Найбільшого розповсюдження отримали наступні підходи та методи [34]:

– точні методи:

1) методи цілочисельного програмування:

- a) лінійне цілочисельне програмування;
- b) нелінійне цілочисельне програмування; булеве програмування;

2) послідовні алгоритми оптимізації:

- a) метод гілок та меж;
- b) динамічне програмування та методи аналізу і відсіву варіантів;
- c) методи теорії графів;

– наближені методи:

1) обмеження об’єму розрахунків у послідовних алгоритмах оптимізації;

2) методи випадкового пошуку:

- a) методи глобального випадкового пошуку,
- b) методи локальної варіації;

3) генетичні алгоритми та еволюційні стратегії;

4) евристичні методи;

5) гібридні алгоритми;

6) метаевристичні методи;

7) імітаційне моделювання.

До наближених методів рішення відносяться методи глобального випадкового пошуку і локальних варіацій [35-38]. Характерною особливістю цього методу є те, що на кожній ітерації процес вибору нового рішення виконується не детерміновано, а в результаті реалізації деякого випадкового процесу, при чому стратегія вибору може залежати від передісторії пошуку. Одним із розвитком методів локальної варіації є Tabu-Search-аналіз [39].

Генетичні алгоритми детально описані у публікаціях [17, 40-47] та є евристичними алгоритмами пошуку, які використовуються для рішення задач оптимізації та моделювання послідовності підбору, комбінації та варіації шуканих параметрів на основі механізмів, які нагадують біологічну еволюцію. Оскільки, алгоритм у процесі пошуку використовує деяке кодування множини параметрів замість самих параметрів, то він може ефективно застосовуватися для рішення задач теорії розкладів, дискретної оптимізації, визначених як на числових множинах, так і на кінцевих множинах вільної природи.

Достатньо широке розповсюдження в наш час отримали різні евристичні методи рішення задачі [19-28]. Основні підходи при побудові цих методів базуються на прийомах зниження вимог та використанні різних видів вирішальних правил, обґрунтованих при рішенні близьких за постановкою задачі, але які не містять цілої множини додаткових складностей та обмежень, характеристик для реально розглянутої проблеми. Евристичні алгоритми використовують різні підходи без суворих обґрунтувань [29, 30, 34]. Евристичні методи дозволяють знаходити прийнятні рішення навіть у дуже складних випадках: при неповноті, випадковості вихідних даних, відсутності адекватної математичної моделі, NP-складності вирішуваної задачі та відсутності точних методів їх рішення.

Також часто для отримання наближеного рішення задачі використовується «лівостороння» схема розгалуження у методі гілок та меж [31, 32] із зупинкою алгоритму у випадку отримання першого допустимого рішення задачі.

Математичні методи вирішення задач теорії розкладів можна поділити на наступні види:

- математичні методи цілочисельного математичного програмування [33, 48-51];
- математичні методи задач теорії розкладів у вигляді побудови допустимих екстремальних шляхів на графі [33, 48, 52-54].

Більшість задач теорії розкладів можуть бути сформульовані як задачі цілочисельного математичного програмування. Частковим випадком яких є задача лінійного цілочисельного програмування [55]. Динамічне програмування [56-60] зазвичай застосовується до вирішенню задач, які можна розбити на послідовність задач меншої розмірності та більш простої структури.

Джонсон запропонував простий алгоритм знаходження точного рішення для задачі обробки n деталей на двох пристроях і для окремого випадку завдання обробки на трьох пристроях, сформульованої Беллманом і званої в даний час завданням Беллмана-Джонсона.

Не зважаючи на те, що більшість задач теорії розкладів є NP-важкі, практика потребує рішення таких задач. Для цього існує декілька підходів. Першим підходом є розробка поліноміальних евристичних алгоритмів. Евристичний алгоритм, або просто евристика — це алгоритм, спроможний видати прийнятне рішення проблеми серед багатьох рішень, але неспроможний гарантувати, що це рішення буде найкращим. Отже, такі алгоритми є приблизними і неточними. Зазвичай такі алгоритми знаходять рішення, близьке до найкращого і роблять це швидко. Іноді такий алгоритм може бути точним, тобто він знаходить дійсно найкраще рішення, але він все одно буде називатися евристичним доти, доки не буде доведено, що рішення дійсно найкраще. Серед евристичних алгоритмів важливу роль відіграють так звані жадібні алгоритми (іноді це носить назву принципу жадності). Цей принцип полягає у тому, щоб на кожному кроці вибирати з усіх можливих альтернатив ту, що забезпечує найменше значення критерію. Цей принцип може бути застосований як при знаходженні початкового розкладу, так і на етапі його дальшого покращення (наприклад, згаданим вище шляхом локального пошуку).

Типовим представником евристичних алгоритмів є метод локального пошуку [61]. При цьому заздалегідь вибрана множина перестановок використовується для послідовного покращення початкового розкладу доти, доки покращення є можливим. Таким чином, знаходиться локальний оптимум у даному класі перестановок.

Іншим напрямком евристичних методів у теорії розкладів формування функцій переваг (пріоритетів). Для кожної i -ї роботи обчислюється значення функції $f(i)$ переваги та вибирається та робота, для якої $f(i)$ досягає мінімуму чи максимуму.

Також евристичним підходом є так звані генетичні алгоритми, суть яких полягає у наступному. Нехай є два допустимі розклади. По цих розкладах за деякими встановленими правилами будується декілька розкладів, кожен з яких є “схожим” у деякому розумінні на ці два вихідні розклади (вони називаються нащадками). З одержаних нащадків вибираються найкращі за заданим критерієм і перевіряється, чи є вони кращими за “батьків”. Якщо так, то по цих розкладах знову будуються нащадки і т.д.

Зустрічаються і такого типу задачі, в яких евристичний алгоритм буде працювати або дуже довго, або видавати невірні результати, однак, такі парадоксальні приклади можуть ніколи не зустрітись на практиці через свою специфічну структуру. Для деяких евристичних алгоритмів відомі оцінки похибки одержуваного рішення. Такі алгоритми називаються наближеними. Існують наближені алгоритми, що гарантують як відносну похибку так і абсолютну похибку. Деякі NP-важкі задачі допускають існування так званої апроксимаційної схеми. В рамках даної схеми можна знайти наближене рішення з відносною похибкою не більше будь-якого заданого значення $\varepsilon > 0$ за час, поліноміально залежний від $1/\varepsilon$ і від розміру вхідної інформації задачі, — цілком поліноміально апроксимаційна схема.

Для задач теорії розкладів такі схеми розробили, наприклад, Алон [62], Мастроліллі [63] та інші. Для завдань, які не мають апроксимаційної схеми, велике значення має встановлення граничного значення ε , для якого можливо знаходження ε -наближеного рішення за поліноміальний час, — поліноміальна апроксимаційна схема.

На даний момент широкого поширення мають метаевристичні алгоритми, які знаходять "гарне" рішення, близьке до оптимального, за прийнятний час. Недоліком таких алгоритмів є відсутність оцінок якості отриманого рішення. Невідомо, наскільки рішення відрізняється від оптимального в найгіршому випадку. Метаевристика є потужним і надзвичайно популярним класом оптимізаційних методів. дозволяють знаходити рішення для широкого кола завдань з різних додатків. Сила метаевристик полягає в їх здатності рішення складних завдань без знання простору пошуку, саме тому ці методи дають можливість вирішувати складні для задачі оптимізації. Спрощено можна назвати метаевристичні алгоритми як алгоритми, що реалізують прямий випадковий пошук можливих рішень задачі, оптимальних або близьких до оптимальних, поки не буде виконана якась умова або досягнуто задане число ітерацій.

Якщо трудомісткість алгоритму рішення задачі дорівнює $O(nk)$, де k – деяка константа, яка не залежить від n , то кажуть, що задача поліноміально вирішувана. Якщо трудомісткість алгоритму залежить від значень числових параметрів задачі, наприклад $O(nA)$ для завдання розбиття, тоді алгоритм псевдо-поліноміальний. Алгоритм трудомісткий виду $O(n^x y^n)$, де x і y – константи, називається експоненціальним. Очевидно, що для деяких n час роботи поліноміального алгоритму може бути більше, ніж час роботи експоненціального алгоритму [64].

Проблеми планування на паралельних пристроях можна класифікувати за такими типами [4]:

- проблема календарного планування на ідентичних паралельних пристроях;
- проблема календарного планування на рівномірних/пропорційних паралельних пристроях;
- проблема календарного планування на не пов'язаних паралельних пристроях.

Нехай, t_{ij} – час обробки роботи j на пристрої i , для $i = 1, 2, 3, \dots, m$ та $j = 1, 2, 3, \dots, n$.

Тоді задаються три способи планування паралельних пристроїв за допомогою цього часу обробки [4].

1) Якщо $t_{ij} = t_{1j}$ для всіх i та j , то проблема називається проблемою, ідентичних паралельних пристроїв. Це означає, що всі паралельні пристрої ідентичні з точки зору їх швидкості. Кожна робота займає однакову кількість часу обробки на кожному з паралельних пристроїв.

2) Якщо $t_{ij} = \frac{t_{1j}}{s_i}$ для всіх i та j , де s_i - швидкість пристрою i , а t_{1j} - час обробки роботи j на пристрої 1, то проблема називається з рівномірним (пропорційним) паралельним пристроєм.

Це означає, що паралельні пристрої матимуть різну швидкість. Як правило, ми припускаємо швидкість s_1, s_2, \dots, s_m для паралельних пристроїв $1, 2, \dots, m$ є відповідно з співвідношенням $s_1 < s_2 < \dots < s_m$. Тобто пристрій 1 є найповільнішим пристроєм, а пристрій m - найшвидшим пристроєм. Для заданої роботи час його обробки на паралельних пристроях буде в співвідношеннях, наведених нижче: $\frac{1}{s_1} : \frac{1}{s_2} : \dots : \frac{1}{s_m}$.

3) Якщо t_{ij} є довільним для всіх i та j , то це проблема незв'язаних паралельних пристроїв. У цьому типі планування не буде ніякого відношення між періодами обробки роботи на паралельних пристроях. Це може бути пов'язано з технологічними відмінностями пристроїв, різними характеристиками робочих місць тощо.

Основні характеристики проблеми календарного планування на рівномірних/пропорційних паралельних пристроях описані нижче [4]:

- є n окремих робіт;
- є m паралельних пристроїв з різною швидкістю ($s_1 < s_2 < \dots < s_m$);
- пристрої постійно доступні, і вони ніколи не залишаються вільними, поки робота чекає;
- t_{1j} - час обробки роботи j на пристрої 1 для $j = 1, 2, 3, \dots, n$;

- для кожної роботи час обробки на однорідних паралельних пристроях обернено пропорційний швидкості цих паралельних пристроях $\left(\frac{1}{s_1} : \frac{1}{s_2} : \dots : \frac{1}{s_m}\right)$,

де s_1 - одиниця швидкості;

- $t_{ij} = \frac{t_{ij}}{s_1}$ for $j=1,2,3,\dots,n$ and $i=1,2,3,\dots,m$;

У роботі розглядається задача складання розкладу виконання робіт з відношенням передування паралельними пристроями (як ідентичних, так і різної продуктивності), тому проаналізуємо сучасні досягнення наукової спільноти у вирішенні задач цього класу із застосуванням різних математичних методів.

2.1.2 Огляд робіт запропонованих методів розв'язку

Оскільки, у роботі будуть проводитися дослідження задач складання розкладів саме для паралельних пристроїв, то проаналізуємо сучасні досягнення щодо вирішення задач цього класу із застосуванням різних математичних методів.

Розроблені огляди літератури про паралельне планування пристроїв наведені в [65] та [66]. Загальновідомо, що проблема мінімізації makespan на паралельних ідентичних пристроях NP-повні навіть з двома пристроями [67-68].

Досягнення різних науковців при вирішенні задачі складання розкладів для паралельних пристроїв детально наведені в роботі [69], порівняльний аналіз деяких таких методів для вирішення makespan задачі складання розкладів наведено в роботі [70] та інших. Для розв'язку задач складання розкладів саме для паралельних пристроїв застосовують такі методи, як випадковий, локальний та вичерпуючий пошук [71-73], метод гілок та меж [74], мурашині алгоритми [75-77], пошук з заборонами [78], ймовірнісні алгоритми [79], генетичні алгоритми [80-87], меметичний алгоритм [88], гібридний алгоритм оптимізації роєм часток [89], алгоритм зозулі [90], 2-наближення [91], бджолиний алгоритм [92], поліноміальний алгоритм [93-95], імітація відпалу [96-99], різні евристичні алгоритми [100-104] та інші.

Також, для розв'язку даної задачі пропонується використання гібридного алгоритму, який поєднує в собі генетичний алгоритм та мурашиний алгоритм [101], змішаного алгоритму нелінійного цілочисельного програмування [105], модифікованого алгоритму поєднання ваги та локального пошуку [106], гібридного алгоритму на основі алгоритму методу рою часток та мурашиного алгоритму [107], застосування нейронної мережі [108].

Проблеми з плануванням широко класифікуються як режими офлайн та онлайн-планування. У офлайн режимі планування час випуску, час обробки, термін виконання та інші необхідні дані кожної з задач. Алгоритми онлайн-планування складають рішення щодо планування кожного разу на основі характеристик робіт, які прибули, не знаючи про роботи, які можуть надходити в майбутньому.

У статті [4] присутній огляд на деякі роботи за даною проблемою.

Проблема календарного планування на ідентичних пристроях формує основи для планування рівномірних паралельних пристроїв. Кофман і Грэхем сформулювали загальну модель обчислювальних структур і навели ефективний алгоритм для пошуку оптимальних non-preemptive розкладів для планування ідентичних паралельних пристроїв з метою мінімізації makespan. Вони довели, що їх алгоритм дає оптимальне рішення.

Горовиць та Сані вперше представили динамічні алгоритми програмування для планування незалежних завдань у багатопроцесорному середовищі, в якому пристрої мають різну швидкість. У цьому дослідженні мета полягає в мінімізації часу закінчення (makespan) та середньозваженого поточного часу на двох процесорах. Ці алгоритми мають найгірші функції складності, які є експоненціальними у кількості завдань. Надалі вони представляли алгоритми наближення низької поліноміальної складності для вищенаведеної задачі.

Ібарра та Кім розробили евристичну програму для планування незалежних завдань на неідентичних пристроях. У їхньому дослідженні, зокрема, для $m = 2$ дано $n \log n$ обмежений за часом алгоритм, який генерує розклад, який має час закінчення

не більше $\frac{(\sqrt{5} + 1)}{2}$ оптимального часу закінчення. Вони підтвердили, що алгоритм LPT, застосований до цієї проблеми, дає розклад, який є близьким до оптимальності для більших n .

Прабудха і Томас Е. Мортон розробили новий евристичний план для роботи на рівномірних паралельних процесорах, щоб мінімізувати makespan. Він протестований на великій кількості робіт для однорідних та ідентичних пристроїв. Вони виявили, що рішення, надані евристикою для рівномірного планування паралельних пристроїв, не перевищують 5% рішень, заданих алгоритмом гілок та меж.

Булфін та Паркер розглянули проблему планування завдань у системі, що складається з двох паралельних пристроїв, мінімізація makespan. Зокрема, вони розглянули різноманітні модифікації цієї основної теми, включаючи випадки ідентичних процесорів, пропорційних і незв'язаних процесорів.

Фрізен та Ленгстон розглянули non-preemptive розклади незалежних робіт до системи однорідних пристроїв з метою зменшення makespan. Відомо, що розклад LPT (найдовший час обробки) у два рази довші за оптимальний makespan.

Ковальов та Шафранський вивчали календарне планування з однорідними пристроями для одиничних робіт з обмеженими ресурсами. Деякі роботи можуть вимагати додаткового ресурсу під час їх виконання. Ресурс відновлюється, але загальне споживання ресурсів обмежене тим самим значенням у кожний момент часу. Вони представили алгоритм $O(m \cdot \log m)$, щоб мінімізувати максимальний час завершення роботи, тобто makespan. Вони також представили лінійний алгоритм часу для залячі з однаковими пристроями для мінімізації максимального часу завершення роботи.

Буркар і Келлерер розробили алгоритм лінійного сполучення для розкладу робіт на рівномірних паралельних пристроях з метою мінімізації makespan. Цей алгоритм має три підпрограми, які запускаються незалежно, для вибору найкращого розподілу між ними.

Чінг-Чонг Ляо та Чіен-Хунг Лін розглянули дві проблеми рівноправних паралельних пристроїв з метою мінімізації makespan. У цій роботі проблема двох однорідних паралельних пристроїв перетворюється на особливу проблему двох однакових паралельних пристроїв з точки зору робочого навантаження замість часу завершення. Розроблено оптимальний алгоритм для трансформованої спеціальної задачі. Запропонований алгоритм має експоненціальну складність часу. Незважаючи на цей факт, автори стверджують, що їх алгоритм може знайти оптимальне рішення для великих задач за короткий час.

Один з шляхів розв'язку задач цього напрямку ґрунтується на застосуванні методології побудови ПДС-алгоритмів для важковирішуваних задач комбінаторної оптимізації.

Зміст ПДС-алгоритмів для важко розв'язувальних задач комбінаторної оптимізації полягає у наступному. Спочатку на основі теоретичного аналізу досліджуваної задачі виявляються ознаки оптимальності допустимих розв'язків, потім розробляється алгоритм розв'язання задачі, що має дві складові поліноміальну і експоненційну. Поліноміальна складова породжується логіко-аналітичними умовами (p -умовами), виконання яких гарантує оптимальність отриманого розв'язку і синтезується таким чином, щоб послідовна процедура конструювання допустимих розв'язків була найбільш ефективна з точки зору реалізації p -умов. Коли допустимий розв'язок, отриманий поліноміальною складовою, не задовольняє p -умовам, то продовжується розв'язання задачі експоненційною складовою алгоритму, чи її поліноміальною апроксимацією [109].

У теорії розкладів на основі цього підходу побудовано ефективні точні ПДС-алгоритми для виконання таких задач:

- мінімізація сумарного запізнення при виконанні незалежних завдань на одному приладі (МСЗ);
- мінімізація сумарного зваженого моменту закінчення виконання частково впорядкованої множини завдань при відношенні порядку, заданому орієнтованим ациклічним графом (МВМ);

– мінімізація сумарного запізнення при виконанні незалежних завдань з рівними директивними термінами паралельними приладами МСЗП) [110].

Дана методологія буде використана для розв'язку задачі складання розкладу виконання робіт з відношенням передування паралельними пристроями шляхом мінімізації загального часу виконання робіт. Наведемо змістовну постановку цієї задачі.

2.2 Змістовна постановка задачі дослідження

Дано плоску конструкцію, на якій можна розміщати предмети, для їх подальшого переміщення (пересування) вантажопідйомним пристроєм (далі *палета*).

Палета складається з S рядів, кожен ряд має H комірок (рис. 2.1).

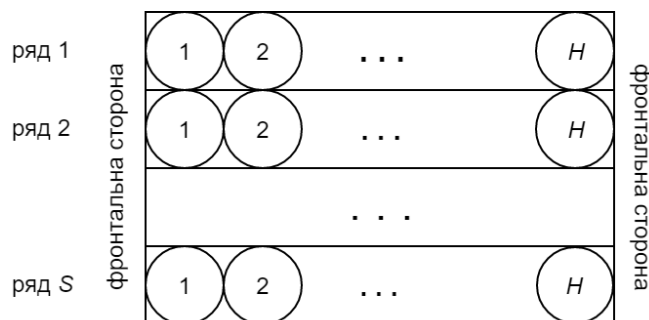


Рисунок 2.1 – Схема палети

Завантаження палети здійснюється тільки зі фронтальних сторін. У загальному випадку кожен ряд палети може завантажуватися з деякої точки, яка розділяє палету на дві частини: перша частина містить q комірок, з номерами $1, 2, \dots, q$, а друга – h комірок, з номерами $q+1, q+2, \dots, H$.

У роботі розглядається два випадки:

- 1) Частковий, де $q = h$ (це можливо тільки за умови $H = 2q$);
- 2) Загальний, де $q \neq h$.

Це означає, що палета ділиться на дві частини (ліву та праву). Одна частина палети містить $S \times q$ комірок – місць для завантаження предметів, інша – $S \times h$ комірок. Порядок завантаження палети для часткового випадку показано на рисунку 2.2, для загального випадку на рисунку 2.3.

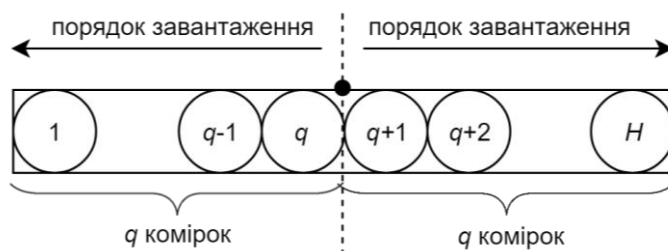


Рисунок 2.2 – Порядок завантаження ряду палети для часткового випадку

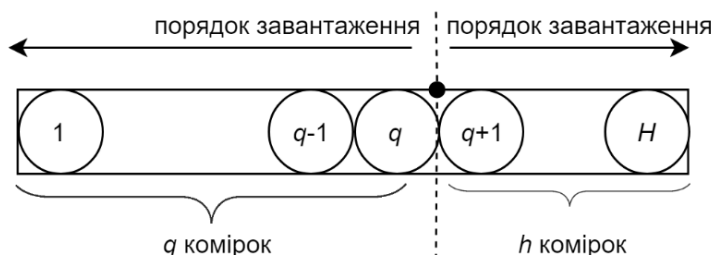


Рисунок 2.3 – Порядок завантаження ряду палети для загального випадку

З урахуванням цього, комірки мають три координати (a, b, c) : a – частина ряду (l – ліва, r – права); b – номер ряду $b = \overline{1, S}$; c – позиція в ряду ($c = \overline{1, q}$ – лівий ряд, $c = \overline{1, h}$ – правий ряд).

На рисунку 2.4 наведено комірки 1-го та 2-го ряду. Комірка, що виділена сірим кольором, має координати $(l, 2, 1)$.

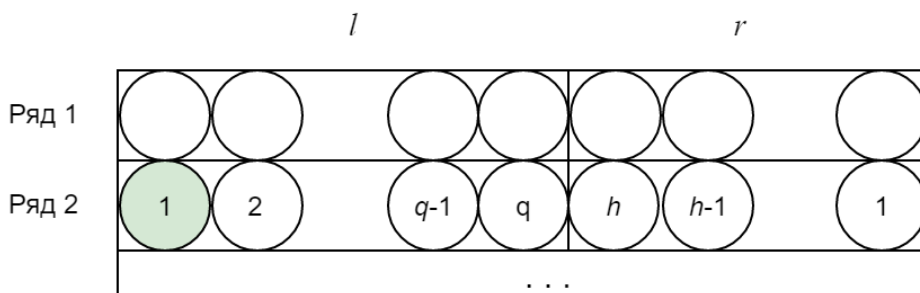


Рисунок 2.4 – Координати комірок

На палету повинно бути завантажено SH предметів, які зберігаються на складі. Комірки в кожному ряді можуть заповнюватися тільки послідовно, починаючи від точки поділу палети на частини, з фронтальних сторін. При цьому, c -та комірка кожного ряду може бути заповнена тільки після того, як в її ряді буде заповнена комірка $c + 1$, ($c = \overline{q - 1, 1}$ – лівий ряд, $c = \overline{h - 1, 1}$ – правий ряд). Палету обслуговують m робітників. Для того, щоб помістити необхідний предмет на позицію abc , роботу j

необхідно витратити $P_{(abc)}j$ одиниць часу $\left(a=l,r;b=\overline{1,n};c=\overline{1,q}\left(c=\overline{1,h}\right),j=\overline{1,m}\right)$. У частковому випадку, коли пристрої мають однакову продуктивність, то $P_{(abc)}j = P_{(abc)}$, $j=\overline{1,m}$. Необхідно визначити порядок завантаження, при якому роботи завантажили б усі комірки за мінімальний час.

Такими чином, задача складання плану завантаження палети зводиться до наступної задачі теорії розкладу: дано SH робіт, що розбиті на n ланцюгів, де $n=2S$. У кожному ланцюзі кожна робота має не більше однієї передуючої і не більше однієї наступуючої роботи, i -тий ланцюг складається з n_i робіт, $i=\overline{1,n}$, p_{ij} – тривалість виконання j -тої роботи i -того ланцюга, $i=\overline{1,n_i}$.

Роботи з відношенням передування i -того ланцюга представлено на рисунку 2.5.



Рисунок 2.5 – Ланцюг робіт з відношенням передування

Дано m пристроїв різної продуктивності, що працюють паралельно. У частковому випадку, коли пристрої мають однакову продуктивність, то $P_{(abc)}j = P_{(abc)}$, $j=\overline{1,m}$.

Метою задачі є визначення такого розкладу виконання робіт множини J , при якому досягає мінімуму загальний час виконання робіт $C_{\max} = \max_{ij} C_{ij} = \max_i C_{in_i} \rightarrow \min$, де C_{ij} – момент завершення роботи ij , C_{in_i} – момент завершення останньої роботи i -того ланцюга.

З огляду на типи пристроїв можна виділити дві задачі визначення мінімуму загального часу виконання робіт із заданим відношенням передування паралельними пристроями:

- задача з ідентичними пристроями;
- задача з неідентичними пристроями.

Перша задача є частковим випадком другої.

Наведемо приклад розкладу для випадку, коли пристрої ідентичні, тобто усі мають однакову продуктивність виконання робіт. Нехай маємо палету, де $S = 3, H = 6$, що опрацьовується двома пристроями з кожної сторони $m = 2$ (рис. 2.6).

	3	2	1	4	5	6
1	4	6	8	4	2	8
2	8	2	6	4	6	4
3	6	4	10	8	4	2

Рисунок 2.6 – Палета з комірками

На рисунку 2.7 наведені ланцюги ($n = 6$) робіт з відношенням передування, що позначено стрілками, які були отримані розділенням робіт точкою поділу палети.

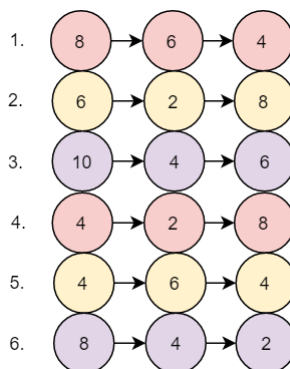


Рисунок 2.7 – Ланцюги робіт з відношенням передування

Тоді один з можливих розкладів виконання робіт може мати вигляд (рис. 2.8).

Пр.1	№1/10	№2/6	№2/2	№3/8	№3/4	
Пр.2	№1/8	№1/6	№2/4	№3/6		
Пр.1	(№3/4)	№4/4	№4/4	№5/2	№6/8	№6/2
Пр.2	№4/8	№5/4	№5/6	№6/4		

Рисунок 2.8 – Розклади робіт при ідентичних пристроях

Для випадку, коли пристрої неідентичні (пропорційні), тобто кожен пристрій має своє значення продуктивності виконання робіт, можливий розклад наведений на рисунку 2.9. Для прикладу, перший пристрій має коефіцієнт продуктивності $k_1 = 1$, другий пристрій має коефіцієнт продуктивності $k_2 = 2$.

Пр.1	№1/10	№1/6	№2/6	№2/2	№3/8	№3/4
Пр.2	№1/16		№2/8		№3/12	
Пр.1	№4/8	№4/4	№5/6	№5/2	№6/8	№6/2
Пр.2	(№3/12)	№4/8	№5/8	№6/8		

Рисунок 2.9 – Розклади робіт при неідентичних пристроях

Висновок до розділу 2

У даному розділі було розглянуто існуючі на сьогоднішній день методи розв'язання задач теорії розкладу та проведений аналіз застосування цих методів для реалізації схожих задач.

У роботі наведено огляд результатів застосування розповсюджених методів розв'язання задач теорії розкладів. При проведенні аналізу літературних джерел було виявлено роботи, що пропонують різні підходи до розв'язання досліджуваної задачі теорії розкладів. Також була описана змістовна постановка задачі. Описаний метод завантаження палети, де кожен ряд палети може завантажуватися з деякої точки, яка розділяє палету на дві частини. Наведений приклад поділу на ланцюги та приклад задачі для неідентичних пристроїв та пристроїв з однаковою продуктивністю.

3 РОЗВ'ЯЗАННЯ ЗАДАЧІ СКЛАДАННЯ РОЗКЛАДУ ВИКОНАННЯ РОБІТ З ВІДНОШЕННЯМ ПЕРЕДУВАННЯ ПАРАЛЕЛЬНИМИ ПРИСТРОЯМИ ЗА КРИТЕРІЄМ МІНІМІЗАЦІЇ ЗАГАЛЬНОГО ЧАСУ ВИКОНАННЯ РОБІТ

Дана задача визначення мінімального загального часу виконання робіт із заданим відношенням передування паралельними пристроями може розглядатися як для ідентичних пристроїв та і для пристроїв, що мають різну продуктивність (рисунок 3.1).

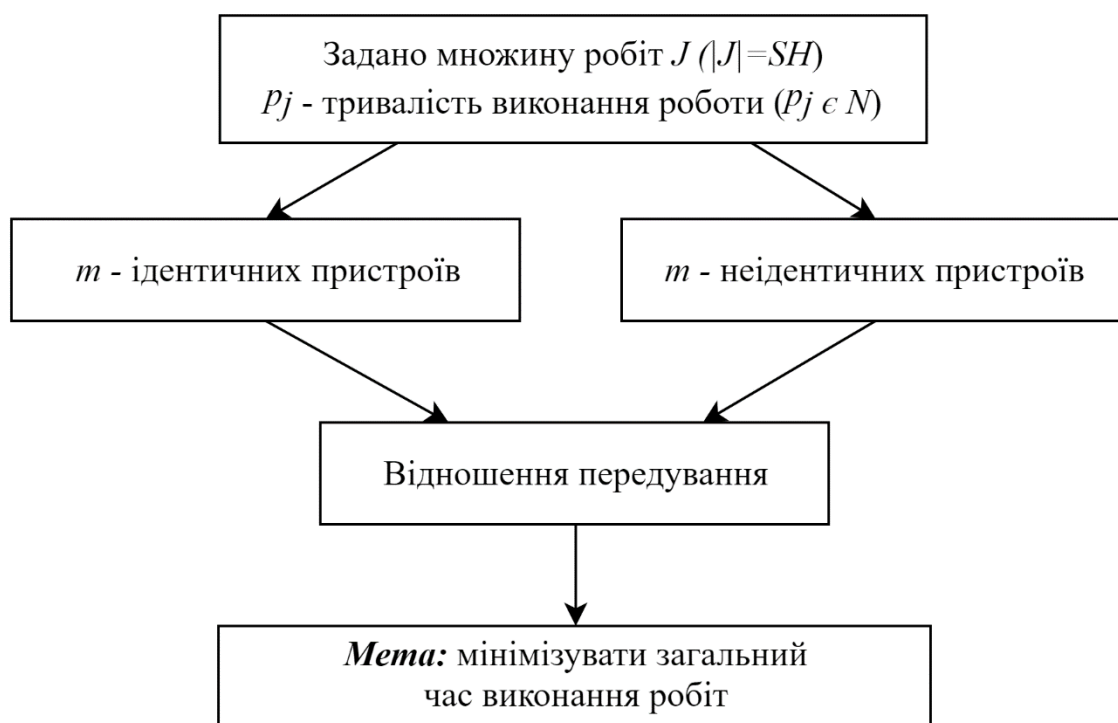


Рисунок 3.1 – Схематичне представлення задач дослідження

Далі у розділі спочатку досліджується задача складання розкладу робіт з відношенням передування паралельними пропорційними пристроями, а потім, як частковий випадок, задача з паралельними ідентичними пристроями. Для цих задач на сьогодні невідомий поліноміальний алгоритм розв'язання.

3.1 Задача складання розкладу виконання робіт з відношенням передування паралельними пристроями різної продуктивності за критерієм мінімізації загального часу виконання робіт

3.1.1 Математична постановка задачі

Задано множину робіт J ($|J| = S \cdot H$), що розбиті на n ланцюгів, де $n = 2S$. Ланцюг i складається з n_i робіт, $i = \overline{1, n}$. Кількість пристроїв, що обробляють дані роботи – m . Кожна робота з J описується парою ij , де i – номер ланцюга, j – порядковий номер у ланцюзі, $j = \overline{1, n_i}$, і тривалістю виконання p_{ij} , $p_{ij} \in N$ (N – множина натуральних чисел). Усі пристрої починають свою роботу в один і той же момент часу (далі – момент запуску пристроїв). Пристрої працюють паралельно і є взаємозамінними у тому сенсі, що кожний з пристроїв може виконувати будь-яку роботу з множини J . Пристрої відрізняються один від одного продуктивністю виконання робіт і є одноманітними (пропорційними). Тобто, пристрої можна впорядкувати за швидкістю виконання робіт і цей порядок однаковий для всіх робіт: для пристрою v заданий коефіцієнт k_v такий, що тривалість виконання роботи ij на пристрої v дорівнює $k_v p_{ij}$, $v = \overline{1, m}$. «Еталонним» будемо називати пристрій з коефіцієнтом продуктивності $k = 1$. У цьому сенсі величина p_{ij} є тривалістю виконання роботи ij на еталонному пристрої. Величину k_v будемо називати *коефіцієнтом продуктивності* (якщо $k_v > 1$, то пристрій v менш продуктивний, ніж еталонний, якщо $k_v < 1$ – більш продуктивний).

Передбачається, що всі роботи множини J надходять одночасно, процес обслуговування кожної роботи протікає без переривань. Усі пристрої працюють без переривань.

Необхідно побудувати такий розклад виконання робіт множини J , при якому досягає мінімуму загальний час виконання робіт: $C_{\max} = \max_{ij} C_{ij} = \max_i C_{in_i} \rightarrow \min$, де C_{ij} – момент завершення роботи ij , C_{in_i} – момент завершення останньої роботи i – того ланцюга.

Далі, не втрачаючи загальності, будемо вважати, що еталонний пристрій є найбільш продуктивний (звідси випливає, що всі $k_v \geq 1$); всі p_{ij} є цілими числами. Задача, що розглядається належить до класу NP.

3.1.2 Дослідження властивостей задачі

Визначимо теоретично мінімальний час, за який усі пристрої могли б виконати всі роботи в об'ємі $\sum_{i=1}^n \sum_{j=1}^{n_i} p_{ij}$. Цю величину можна отримати, якщо вважати, що усі пристрої можуть паралельно виконувати кожну з робіт. Для цього m паралельних пристроїв розглядаються як один, а фактичні тривалості робіт замінюються так званими узагальненими значеннями, рівними:

$$\tilde{p}_{ij} = \frac{1}{\sum_{v=1}^m \frac{1}{t_{vij}}},$$

де $t_{vij} = k_v p_{ij}$ – тривалість виконання роботи ij пристроєм v ($v = \overline{1, m}$; $j = \overline{1, n_i}$);

$\frac{1}{t_{vij}}$ – частка роботи ij , яка виконується пристроєм v за одну одиницю часу;

$\sum_{v=1}^m \frac{1}{t_{vij}}$ – частка роботи ij , яка виконуються всіма пристроями за одну одиницю

часу;

$\frac{1}{\sum_{v=1}^m \frac{1}{t_{vij}}}$ – тривалість виконання роботи ij всіма пристроями.

Тоді, з однієї сторони, мінімально можливий час, за який усі пристрої могли б виконати усі роботи, відповідає «рівномірному» розкладу, тобто розкладу, в якому кожен з пристроїв закінчує роботу в момент:

$$\sum_{i=1}^n \sum_{j=1}^{n_i} p_{ij},$$

$$\sum_{i=1}^n \sum_{j=1}^{n_i} \frac{1}{\sum_{v=1}^m \frac{1}{k_v p_{ij}}}. \quad (3.1)$$

Однак, з урахуванням того, що в даній задачі роботи мають відношення передування у вигляді ланцюгів, то загальний час на виконання усіх робіт не може бути меншим, ніж час виконання робіт найдовшого з ланцюгів найпродуктивнішим пристроєм. Тобто нехай $P_i = \sum_{j=1}^{n_i} p_{ij}$, $i = \overline{1, n}$. Упорядкуємо ланцюги по незростанню їх сумарної тривалості робіт $P_1 \geq P_2 \geq \dots \geq P_n$, а пристрої за їх продуктивністю так, що $k_1 \leq k_2 \leq \dots \leq k_m$, $k = \overline{1, m}$. Тоді нижня оцінка критерію C^* з урахуванням (3.1):

$$C^* = \max \left\{ P_1 k_1; P_2 k_2; \dots; P_{m-1} k_{m-1}; \sum_{i=1}^n \sum_{j=1}^{n_i} \tilde{p}_{ij} \right\}$$

$$C^* = \max \left\{ P_1 k_1; P_2 k_2; \dots; P_{m-1} k_{m-1}; \sum_{i=1}^n \sum_{j=1}^{n_i} \frac{1}{\sum_{v=1}^m \frac{1}{k_v p_{ij}}} \right\} \quad (3.2)$$

Величина C^* може бути як цілим так і не цілим числом.

Враховуючи (3.2), маємо два випадки, що є важливими для аналізу задачі:

- 1) коли максимум у (3.2) відповідає останньому елементу;
- 2) коли максимум у (3.2) відповідає одному з перших елементів.

Розглянемо деякий розклад σ . Позначимо в цьому розкладі:

$J_v(\sigma)$ – упорядкована множина робіт, що виконується пристроєм v , $v = \overline{1, m}$;

$C_v(\sigma) = \sum_{ij \in J_v(\sigma)} k_v p_{ij}$ – тривалість зайнятості пристрою v , $v = \overline{1, m}$;

$\Delta_v(\sigma) = \max \{0; C_v(\sigma) - C^*\} = \max \left\{ 0; \sum_{j \in J_v(\sigma)} k_v p_{ij} - C^* \right\}$ – *виступ* пристрою v , $v = \overline{1, m}$;

$R_v(\sigma) = \max \{0; C^* - C_v(\sigma)\} = \max \left\{ 0; C^* - \sum_{j \in J_v(\sigma)} k_v p_{ij} \right\}$ – *резерв* пристрою v , $v = \overline{1, m}$.

Позначимо через $C_v^{\prime}(\sigma) = \frac{C_v(\sigma)}{k_v}$ зведену тривалість зайнятості пристрою v

($v = \overline{1, m}$) в розкладі σ – час зайнятості пристрою, розрахований у еталонних

тривалостях виконання робіт. Відповідні зведені величини:

$$\Delta'_v(\sigma) = \max \left\{ 0; C'_v(\sigma) - c_v^* \right\}; \quad v = \overline{1, m}; \quad R'_v(\sigma) = \max \left\{ 0; c_v^* - C'_v(\sigma) \right\}, \quad v = \overline{1, m}.$$

Визначимо:

$I_\Delta(\sigma)$ – множина пристроїв, для яких $\Delta_v(\sigma) > 0$;

$I_R(\sigma)$ – множина пристроїв, для яких $R_v(\sigma) > 0$;

$I_0(\sigma)$ – множина пристроїв, для яких $\Delta_v(\sigma) = R_v(\sigma) = 0$.

Із визначення величин $R_v(\sigma)$ і $\Delta_v(\sigma)$ слідує, що $R_v(\sigma)\Delta_v(\sigma) = 0$, $v = \overline{1, m}$.

З урахуванням позначень критерій оптимізації вихідної задачі (мінімізація загального часу виконання робіт) має вигляд:

$$C_{\max}(\sigma) = C^* + \max_{1 \leq v \leq m} \Delta_v(\sigma) \rightarrow \min$$

і є еквівалентним критерію мінімізації загального часу виконання всіх робіт:

$$C^* + \max_{1 \leq v \leq m} \Delta_v(\sigma) \rightarrow \min.$$

У свою чергу, з урахуванням того, що $C^* = \text{const}$, тобто не залежать від упорядкування, критерій спрощується і зводиться до мінімізації максимального виступу:

$$C_{\max}(\sigma) \rightarrow \min \sim \max_{1 \leq v \leq m} \Delta_v(\sigma) \rightarrow \min.$$

При побудові розкладів будемо вважати, що момент r запуску пристроїв дорівнює нулю. Рисунок 3.2 ілюструє взаємозв'язок величин, що входять до вказаних критеріїв.

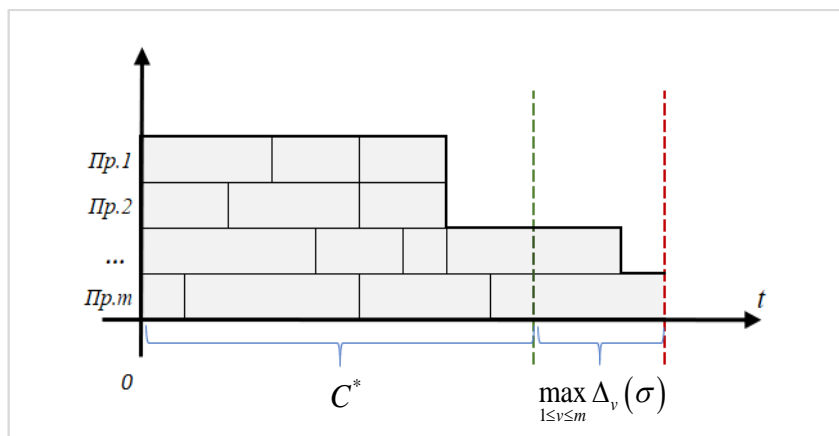


Рисунок 3.2 – Взаємозв'язок величин

Для введених величин має місце наступне:

$$\sum_{ij \in J_v(\sigma)} k_v p_{ij} = C^* - R_v(\sigma) + \Delta_v(\sigma), \quad v = \overline{1, m}. \quad (3.3)$$

Розділимо v -те рівняння в (3.3) на k_v ($k_v \neq 0$):

$$\sum_{ij \in J_v(\sigma)} p_{ij} = \frac{C^*}{k_v} - \frac{R_v(\sigma)}{k_v} + \frac{\Delta_v(\sigma)}{k_v}, \quad v = \overline{1, m},$$

$$\sum_{ij \in J_v(\sigma)} p_{ij} = c_v^* - R'_v(\sigma) + \Delta'_v(\sigma), \quad v = \overline{1, m}.$$

Застосовуючи методологію побудови ПДС-алгоритмів [11], згідно з якою поліноміальна складова алгоритму породжується логіко-аналітичними умовами, виконання яких гарантує оптимальність отриманого рішення, визначимо достатні умови оптимальності (ДУО) розкладів.

3.1.2.1 Достатні умови оптимальності розкладів

Сформулюємо ДУО розкладів для двох випадків.

Випадок 1

Як було вказано раніше, «ідеальним» розкладом є рівномірний розклад, тобто розклад, у якого $\sum_{v=1}^m \Delta_v(\sigma) = 0$; $\sum_{v=1}^m R_v(\sigma) = 0$. Рівномірний розклад можливо отримати

тільки тоді при $C^* = \sum_{i=1}^n \sum_{j=1}^{n_i} \frac{1}{\sum_{v=1}^m \frac{1}{k_v p_{ij}}}$ і $C_{\max}(\sigma) = C^*$, коли C^* та усі $c_v^* = \frac{C^*}{k_v}$ є цілими числами

(інакше не можливі рівності $\sum_{ij \in J_v} p_{ij} = c_v^*$, $v = \overline{1, m}$ – ліві частини рівностей за припущенням є сумами цілих величин).

Випадок 2

Якщо $C^* = \max\{P_1 k_1; P_2 k_2; \dots; P_{m-1} k_{m-1}\}$ та $C_{\max}(\sigma) = C^*$ то розклад σ є оптимальним. При цьому $\sum_{v=1}^m \Delta_v(\sigma) = 0$.

Контур оптимального розкладу для двох випадків наведено на рисунку 3.3.

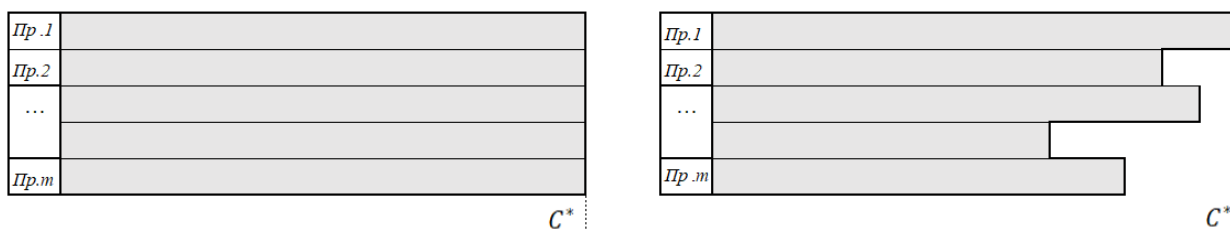


Рисунок 3.3 – Контур ідеального розкладу

Далі визначимо етапи процесу розв'язання вихідної задачі.

3.1.3 Процес розв'язання вихідної задачі

Процес розв'язання вихідної задачі знаходження розкладу робіт складається з двох етапів:

Етап 1. Підготовка:

- 1.1 визначення точки поділу палети;
- 1.2 формування ланцюгів робіт.

Етап 2. Побудова розкладу:

- 2.1 побудова початкового розкладу;
- 2.2 покращення результатів.

3.1.3.1 Знаходження точки поділу палети

На першому етапі процесу розв'язання вихідної задачі початковим кроком є знаходження точки поділу палети, яка розділить роботи на 2 ланцюги в кожному ряді. Точка поділу може бути довільною.

Алгоритм для пошуку точок поділу палети має наступні чотири кроки.

Алгоритм

КРОК 1 Ділимо поточний ряд на дві рівні частини. Фіксуємо точку поділу ряду.

Отримаємо 2 ланцюги.

КРОК 2 Рахуємо загальний час виконання всіх робіт у кожному ланцюгу.

КРОК 2.1 Рахуємо центральний загальний час виконання всіх робіт у кожному ланцюгу.

КРОК 2.2 Переносимо точку поділу на одну позицію вправо та рахуємо правий загальний час виконання всіх робіт у кожному ланцюгу.

КРОК 2.3 Переносимо точку поділу на одну позицію вліво та рахуємо лівий загальний час виконання всіх робіт у кожному ланцюгу.

КРОК 3 Рахуємо різницю часу виконання робіт ланцюгів.

КРОК 3.1 Рахуємо центральну різницю між загальним часом виконання робіт ланцюгів по модулю.

КРОК 3.2 Рахуємо праву різницю загального часу виконання робіт ланцюгів по модулю.

КРОК 3.3 Рахуємо ліву різницю загального часу виконання робіт ланцюгів по модулю.

КРОК 4 Обираємо точку поділу.

КРОК 4.1 Порівнюємо різниці загального часу виконання робіт та обираємо різницю, яка є найменшою (одну з найменших).

КРОК 4.2 Якщо мінімальна сума не є центральною різницею загального часу виконання робіт – ставимо мінімальну різницю як центральну та переходимо на **КРОК 3.2**.

КРОК 4.3 Фіксуємо вибрану точку як точку поділу поточного ряду.

КРОК 5 Якщо ряд не є останнім, ставимо наступний ряд як поточний та переходимо на **КРОК 1**.

КІНЕЦЬ

Приклад застосування алгоритму

Нехай маємо палету, де $S = 3, H = 6$ (рисунок 3.4). Кожна комірка представляє роботу, що має свій час виконання.

1	4	6	8	4	2	8
2	8	2	6	4	6	4
3	6	4	10	8	4	2

Рисунок 3.4 – Палета

Для першого ряду знайдемо точку поділу:

КРОК 1 Ділимо поточний ряд на дві рівні частини. Фіксуємо точку поділу ряду.

Отримаємо 2 ланцюги (рисунок 3.5).

КРОК 2 Рахуємо загальний час виконання всіх робіт у кожному

ланцюгу (рисунок 3.6).

КРОК 2.1 Центральний загальний час виконання всіх робіт у першому ланцюгу: $4+6+8=18$; у другому ланцюгу: $4+8+8=20$.

КРОК 2.2 Правий загальний час виконання всіх робіт у першому ланцюгу: $4+6+8+4=22$ у другому ланцюгу: $8+8=16$.

КРОК 2.3 Лівий загальний час виконання всіх робіт у першому ланцюгу: $4+6=10$ у другому ланцюгу: $8+4+8+8=28$.

КРОК 3 Рахуємо різницю часу виконання робіт ланцюгів.

КРОК 3.1 Центральна різниця: $|18 - 20| = 2$.

КРОК 3.2 Права різниця: $|22 - 16| = 6$.

КРОК 3.3 Ліва різниця: $|10 - 28| = 18$.

КРОК 4 Обираємо точку поділу (рисунок 3.6).

КРОК 4.1 Порівнюємо різниці: $2 < 6 < 18$.

Обираємо центральну різницю: 2.

КРОК 4.3 Фіксуємо вибрану точку як точку поділу поточного ряду.

Аналогічні дії проводяться для кожного рядка палети.

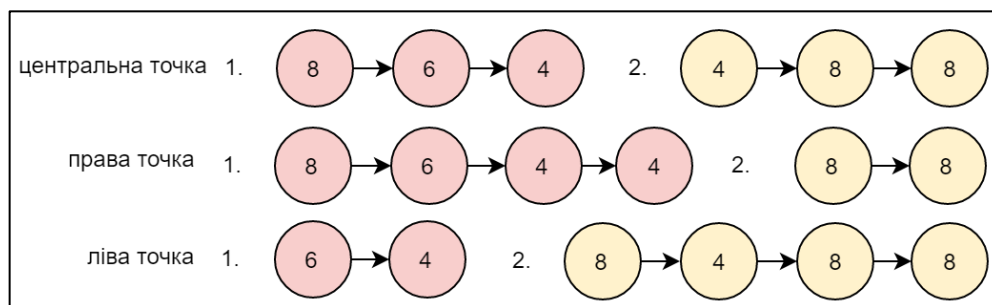


Рисунок 3.5 – Розбиття на ланцюги точками поділу

1	4	6	8	4	8	8
2	8	2	6	4	6	4
3	6	4	10	8	4	2

Рисунок 3.6 – Знаходження точок поділу палети

Після знаходження точок поділу палети формуємо ланцюги, які є результатом поділу кожного з ряду палети роздільною точкою. Кожен з n ланцюгів має набір робіт зв'язаних відношенням передування. Тобто, кожна наступна робота у ланцюзі

може бути виконана пристроєм тільки після виконання попередньої.

3.1.3.2 Побудова розкладу виконання робіт

Другий етап починається з побудови початкового розкладу виконання робіт.

3.1.3.2.1 Побудова початкового розкладу

Розроблено чотири евристичних алгоритми побудови початкового розкладу, метою яких є рівномірне навантаження пристроїв.

Алгоритм А1 – алгоритм побудови початкового розкладу без простоїв

У даному алгоритмі поточна робота призначається на той пристрій, в якого невикористаний зведений фонд робочого часу найбільший і попередня робота завершена до часу завершення поточної останньої роботи цього пристрою.

Позначення: $J^* \in J$ – множина робіт, в яких усі попередні роботи призначені на виконання.

КРОК 1. Розрахувати C^* .

КРОК 2. Сформувані множини $J^* := \{p_{11}; p_{21}; \dots; p_{n1}\}$.

КРОК 3. Встановити значення величин невикористаного зведеного фонду

робочого часу пристроїв: $f_v := \frac{C^*}{k_v}$, $v = \overline{1, m}$.

КРОК 4. Встановити поточну тривалість зайнятості пристроїв: $T_v := 0$, $v = \overline{1, m}$.

КРОК 5. Обрати серед робіт множини J^* , роботу ij , що має найбільшу тривалість.

КРОК 6. Визначити множину пристроїв $M^* \in M$, на які можна призначити роботу ij : $v \in M^*$, якщо $T_v \geq C_{i,j-1}$ (робота, що передуює роботі ij виконана до моменту поточної тривалості зайнятості пристрою).

КРОК 7. Обрати з множини M^* пристрій v із максимальним невикористаним зведеним фондом робочого часу f_v (а якщо таких декілька, то обрати пристрій з найменшим k_v).

КРОК 8. Призначити роботу ij на пристрій v : $T_v := T_v + k_v p_{ij}$; $f_v := f_v - p_{ij}$.

КРОК 9. Видалити роботу ij з множини J^* : $J^* := J^* \setminus \{(i, j)\}$.

ЯКЩО $j < n_i$, **ТО** $J^* := J^* \cup \{(i, j+1)\}$.

КРОК 10. **ЯКЩО** $J^* = \emptyset$ **ТО** кінець алгоритму,

ІНАКШЕ перейти на **КРОК 5**.

Алгоритм А2 – алгоритм побудови початкового розкладу з простоями

У даному алгоритмі поточна робота призначається на той пристрій, у якого невикористаний зведений фонд робочого часу найбільший, але після моменту, коли попередня робота була призначена.

Позначення: B_{ij} – початок виконання роботи ij .

КРОК 1. Розрахувати C^* .

КРОК 2. Сформувати множину робіт $J^* := \{p_{11}; p_{21}; \dots; p_{n1}\}$.

КРОК 3. Встановити значення величин невикористаного зведеного фонду

робочого часу пристроїв: $f_v := \frac{C^*}{k_v}$, $v = \overline{1, m}$.

КРОК 4. Встановити поточну тривалість зайнятості пристроїв: $T_v := 0$, $v = \overline{1, m}$.

КРОК 5. Обрати серед робіт множини J^* , роботу ij , що має найбільшу тривалість.

КРОК 6. Обрати пристрій v з множини M із максимальним невикористаним зведеним фондом робочого часу f_v (а якщо таких декілька, то обрати пристрій з найменшим k_v).

КРОК 7. Призначити роботу ij на пристрій v :

ЯКЩО $T_v \geq C_{i,j-1}$; **ТО** $B_{ij} = T_v$; $T_v := T_v + k_v p_{ij}$; $f_v = f_v - p_{ij}$;

ІНАКШЕ $B_{ij} := C_{i,j-1}$; $T_v := C_{i,j-1} + k_v p_{ij}$; $f_v = f_v - (C_{i,j-1} + k_v p_{ij}) / k_i$.

КРОК 8. Видалити роботу ij з множини J^* : множини $J^* = J^* \setminus \{(i, j)\}$.

ЯКЩО $j < n_i$, **ТО** $J^* = J^* \cup \{(i, j+1)\}$,

ЯКЩО $J^* = \emptyset$ **ТО** кінець алгоритму,

ІНАКШЕ перейти на **КРОК 5**.

Алгоритм А3 – алгоритм побудови розкладу випадковим чином без простоїв

У даному алгоритмі поточна робота призначається на випадковий пристрій з множини пристроїв, для яких попередня робота була завершена до часу завершення останньої роботи цього пристрою.

Позначення: $J^* \in J$ – множина робіт, в яких усі попередні роботи призначені на виконання.

КРОК 1. Розрахувати C^* .

КРОК 2. Сформуванати множину $J^* := \{p_{11}; p_{21}; \dots; p_{n1}\}$.

КРОК 3. Встановити значення величин невикористаного зведеного фонду робочого часу пристроїв: $f_v := \frac{C^*}{k_v}$, $v = \overline{1, m}$.

КРОК 4. Встановити поточну тривалість зайнятості пристроїв: $T_v := 0$, $v = \overline{1, m}$.

КРОК 5. Обрати серед робіт множини J^* , роботу ij , що має найбільшу тривалість.

КРОК 6. Визначити множину пристроїв $M^* \in M$, на які можна призначити роботу ij : $v \in M^*$, якщо $T_v \geq C_{i,j-1}$ (робота, що передуює роботі ij виконана до моменту поточної тривалості зайнятості пристрою).

КРОК 7. Обрати з множини M^* будь-який пристрій v .

КРОК 8. Призначити роботу ij на пристрій v : $T_v := T_v + k_v p_{ij}$; $f_v := f_v - p_{ij}$.

КРОК 9. Видалити роботу ij з множини J^* : $J^* := J^* \setminus \{(i, j)\}$.

ЯКЩО $j < n_i$, **ТО** $J^* := J^* \cup \{(i, j+1)\}$.

КРОК 10. ЯКЩО $J^* = \emptyset$ **ТО** кінець алгоритму,

ІНАКШЕ перейти на **КРОК 5**.

Алгоритм А4 – алгоритм побудови розкладу випадковим чином з простоями

У даному алгоритмі поточна робота призначається на той пристрій, у якого невикористаний зведений фонд робочого часу найбільший, але після моменту, коли попередня робота була призначена.

Позначення: B_{ij} – початок виконання роботи ij .

КРОК 1. Розрахувати C^* .

КРОК 2. Сформувати множину робіт $J^* := \{p_{11}; p_{21}; \dots; p_{n1}\}$.

КРОК 3. Встановити значення величин невикористаного зведеного фонду робочого часу пристроїв: $f_v := \frac{C^*}{k_v}$, $v = \overline{1, m}$.

КРОК 4. Встановити поточну тривалість зайнятості пристроїв: $T_v := 0$, $v = \overline{1, m}$.

КРОК 5. Обрати серед робіт множини J^* , роботу ij , що має найбільшу тривалість.

КРОК 6. Обрати будь-який пристрій v з множини M .

КРОК 7. Призначити роботу ij на пристрій v :

ЯКЩО $T_v \geq C_{i,j-1}$; **ТО** $B_{ij} = T_v$; $T_v := T_v + k_v p_{ij}$; $f_v = f_v - p_{ij}$;

ІНАКШЕ $B_{ij} := C_{i,j-1}$; $T_v := C_{i,j-1} + k_v p_{ij}$; $f_v = f_v - (C_{i,j-1} + k_v p_{ij})/k_i$.

КРОК 8. Видалити роботу ij з множини J^* : множини $J^* = J^* \setminus \{(i, j)\}$.

ЯКЩО $j < n_i$, **ТО** $J^* = J^* \cup \{(i, j+1)\}$,

ЯКЩО $J^* = \emptyset$ **ТО** кінець алгоритму,

ІНАКШЕ перейти на **КРОК 5**.

Наступним кроком є розробка множини операцій обміну.

3.1.3.2.2 Розробка множини операцій обміну

Отже, маємо контур оптимального розкладу для заданої сумарної довжини робіт множини J , у якого v -й пристрій завершує свою множину робіт до моменту: C^* , якщо $\delta \leq 0$; C_v^* , якщо $\delta > 0$ (рисунок 3.7).

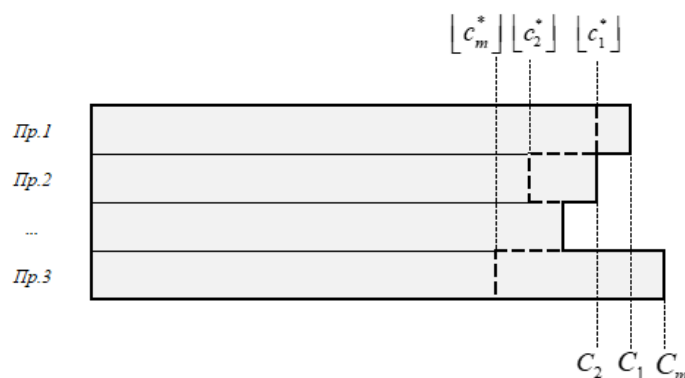


Рисунок 3.7 – Оптимальний контур розкладу

Для покращення розкладу необхідно направити зусилля на зменшення величини $\max_v \Delta_v(\sigma)$. Для цього пропонується використовувати обмін робіт між двома пристроями: перший пристрій h з множини $I_\Delta(\sigma)$ з максимальним значенням $\max_{v \in I_\Delta} \Delta_v(\sigma)$, а другий – довільний пристрій s із множини $I_R(\sigma), I_\Delta(\sigma), I_0(\sigma), (s \neq h)$. При цьому, деяка робота з пристроєм h ($ij \in J_h(\sigma)$) міняється місцями з деякою роботою з пристроєм s ($kl \in J_s(\sigma)$). Далі такий процес будемо називати операцією обміну робіт між підмножинами $J_h(\sigma)$ та $J_s(\sigma)$.

Позначимо через θ різницю між еталонними тривалостями робіт, які приймають участь в обміні:

$$\theta = p_{ij} - p_{kl}.$$

Нехай σ^1 – розклад, який отримано після виконання операції обміну.

Для того, щоб перестановка призвела до покращення (тобто $\sum_{v=1}^m \Delta_v(\sigma) - \sum_{v=1}^m \Delta_v(\sigma^1) > 0$), необхідно, щоб $\theta > 0$.

У роботі розроблена множина операцій обміну, які в залежності від умов виконання та наслідків розділено на чотири типи: A^k, B^k, C^k та Γ^k .

У даній задачі обмін може бути можливим тільки за умови, що відношення передування робіт на всіх пристроях після операції обміну буде збережено. Тобто при обміні, між роботою ij пристроєм $J_h(\sigma)$ та роботою kl пристроєм $J_s(\sigma)$, множина робіт $J'_h(\sigma)$, що від ij і до останньої роботи пристроєм $J_h(\sigma)$ та множина робіт $J'_s(\sigma)$, що від kl і до останньої роботи пристроєм $J_s(\sigma)$ обов'язково задовольняють наступним умовам. Для прикладу візьмемо роботу ij пристроєм $J_h(\sigma)$ і kl пристроєм $J_s(\sigma)$:

$$1. C_{(i,j-1)} \leq C_{kl} - p_{kl} \cdot k_s \text{ і } C_{(k,l-1)} \leq C_{ij} - p_{ij} \cdot k_h;$$

$$2. \text{ Якщо } C_{(i,j-1)} \in J_s,$$

$$\text{то при } C_{(i,j-1)} \leq C_{kl} - p_{kl} \cdot k_s : C_{(i,j-1)} \leq C_{i,j} - p_{ij} \cdot k_h - \theta \cdot k_h,$$

$$\text{а при } C_{(i,j-1)} > C_{kl} - p_{kl} \cdot k_s : C_{(i,j-1)} + \theta \cdot k_s \leq C_{(i,j)} - p_{ij} \cdot k_h - \theta \cdot k_h;$$

3. Якщо $C_{(k,l-1)} \in J_h$,

то при $C_{(k,l-1)} \leq C_{ij} - p_{ij} \cdot k_h$: $C_{(k,l-1)} \leq C_{kl} - p_{kl} \cdot k_s + \theta \cdot k_s$,

а при $C_{(k,l-1)} > C_{ij} - p_{ij} \cdot k_h$: $C_{(k,l-1)} - \theta \cdot k_h \leq C_{(k,l)} - p_{kl} \cdot k_s + \theta \cdot k_s$.

При всіх операціях обміну дана умова є обов'язковою. Надалі, щоб уникнути дублювання вона буде опущена.

Мета операцій обміну типу A^k : зменшення максимального з виступів $\Delta_h(\sigma) = \max_{v \in I_\Delta(\sigma)} \Delta_v(\sigma)$ за рахунок зменшення резерву одного з пристроїв множини $I_R(R_v(\sigma), v \in I_R)$. Умови виконання операції обміну типу A^k :

$$\theta > 0,$$

$$\theta \cdot k_h \leq \Delta_h(\sigma), \quad \theta \cdot k_s \leq R_s(\sigma). \quad (3.4)$$

У результаті обміну отримуємо розклад σ^1 , у якого:

$$\Delta_h(\sigma^1) = \Delta_h(\sigma) - \theta \cdot k_h, \quad R_s(\sigma^1) = R_s(\sigma) - \theta \cdot k_s.$$

Виступ, що був максимальним у σ , зменшився на $\theta \cdot k_h$:

$$\Delta_h(\sigma^1) = \max_{v \in I_\Delta(\sigma)} \Delta_v(\sigma) - \theta \cdot k_h.$$

Таким чином, розклад σ^1 не гірший за розклад σ , а за відсутності альтернатив при виборі пристрою h – кращий за σ .

При обміні міняються місцями робота ij з пристроєм $h \in I_\Delta(\sigma)$ та робота kl з пристроєм $s \in I_R(\sigma)$, робота ij та kl задовольняють наступним умовам:

$$\theta = p_{ij} - p_{kl} > 0,$$

$$(p_{ij} - p_{kl}) \cdot k_h \leq \Delta_h(\sigma), \quad (3.5)$$

$$(p_{ij} - p_{kl}) \cdot k_s \leq R_s(\sigma). \quad (3.6)$$

У результаті обміну отримуємо розклад σ^1 :

$$\Delta_h(\sigma^1) = \Delta_h(\sigma) - (p_{ij} - p_{kl}) \cdot k_h, \quad R_s(\sigma^1) = R_s(\sigma) - (p_{ij} - p_{kl}) \cdot k_s.$$

Рисунок 3.8-а) ілюструє обмін A^k , для якої (3.6) виконуються як строгі нерівності, а на рисунку 3.8-б) показано перестановку A^k , у якої передумова (3.5) виконується як строга нерівність, а (3.6) – як строга рівність. На рисунках цього

підрозділу та додатку A на рисунках А.1-А.4 пунктирною лінією позначено значення C^* , відносно якого розраховуються виступи і резерви пристроїв.

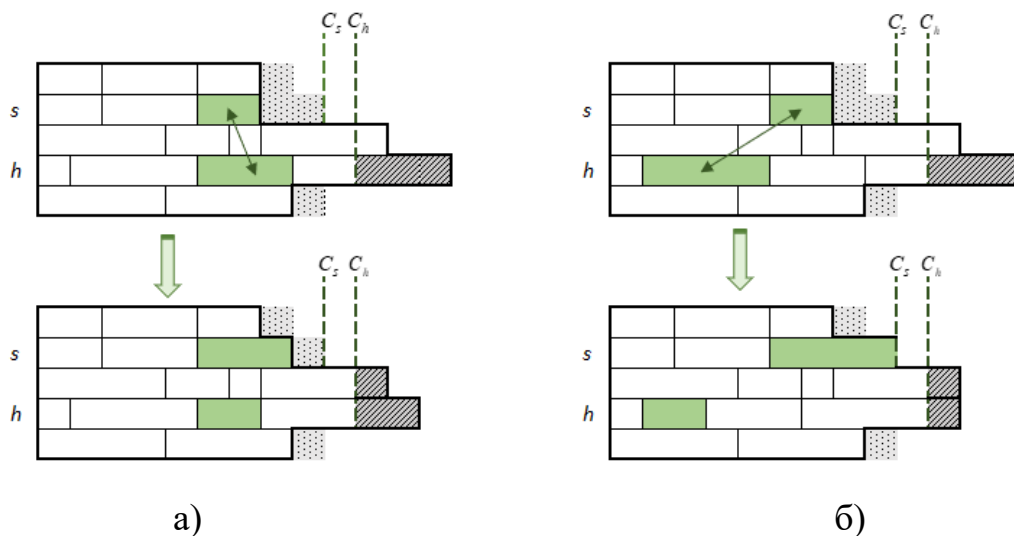


Рисунок 3.8 – Обміни типу A^k

Узагальнена характеристика та властивості обмінів типу A^k представлені в таблиці 3.2. У результаті обмінів типу A^k потужності множин I_Δ та I_R залишаються незмінними або зменшуються. Проте, ця множина обмінів в деяких ситуаціях не дозволяє покращити поточний розв'язок (наприклад, розклад, який наведено на рисунку 3.9 не підлягає оптимізації множиною обмінів типу A^k). Отже, є необхідність розробки операцій обмінів нового типу.

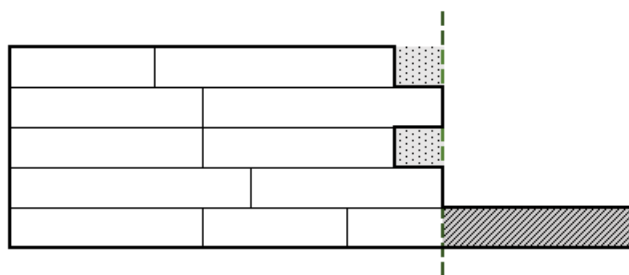


Рисунок 3.9 – Приклад обмежених можливостей обмінів типу A^k

Мета операцій обміну типу B^k : зменшення максимального з виступів за рахунок резерву пристрою з множини I_R і з утворенням виступу на цьому пристрої. У результаті обмінів цього типу може збільшитись множина I_Δ .

Умови виконання обміну типу B^k :

$$\theta > 0, \quad \theta \cdot k_h \leq \Delta_h(\sigma), \quad \theta \cdot k_s > R_s(\sigma).$$

У результаті обміну отримуємо розклад σ^1 , в якого:

$$\Delta_h(\sigma^1) = \Delta_h(\sigma) - \theta \cdot k_h, \quad \Delta_s(\sigma^1) = \theta \cdot k_s - R_s(\sigma) > 0, \\ s \in I_\Delta(\sigma^1).$$

Виступ, що був максимальним у σ , зменшився на $\theta \cdot k_h$:

$$\Delta_h(\sigma^1) = \max_{i \in I_\Delta(\sigma)} \Delta_i(\sigma) - \theta \cdot k_h = \Delta_h(\sigma) - \theta \cdot k_h,$$

але при цьому для новоутвореного виступу $\Delta_s(\sigma^1)$ виконується: $\Delta_s(\sigma^1) < \Delta_h(\sigma)$.

Покажемо це: $\Delta_h(\sigma) - \Delta_s(\sigma^1) = \Delta_h(\sigma) - (\theta \cdot k_s - R_s(\sigma)) = \Delta_h(\sigma) + R_s(\sigma) - \theta \cdot k_s > 0$,

оскільки, $\Delta_h(\sigma) - \theta \cdot k_h \geq 0$, $R_s(\sigma) > 0$. Таким чином, розклад σ^1 не гірший за розклад σ ,

а за відсутності альтернатив при виборі пристрою h – кращий за σ .

Обмін підтипу B^k : міняються місцями робота ij з пристрою $h \in I_\Delta(\sigma)$ та робота kl з пристрою $s \in I_R(\sigma)$, роботи ij та kl задовольняють наступним умовам:

$$\theta = p_{ij} - p_{kl} > 0,$$

$$(p_{ij} - p_{kl}) \cdot k_h \leq \Delta_h(\sigma),$$

$$(p_{ij} - p_{kl}) \cdot k_s > R_s(\sigma).$$

Для отриманого в результаті обміну розкладу σ^1 виконується:

$$\Delta_h(\sigma^1) = \Delta_h(\sigma) - (p_{ij} - p_{kl}) \cdot k_h,$$

$$\Delta_s(\sigma^1) = (p_{ij} - p_{kl}) \cdot k_s - R_s(\sigma).$$

Приклад ілюстрації обміну наведений у додатку А, рисунок А.2. Узагальнена характеристика та властивості обмінів типу B^k представлені в таблиці 2.2.

Множина обмінів типу B^k призводить до того, що поточна потужність множини I_Δ збільшується, але при цьому величина максимального з виступів зменшується. Наступна множина обмінів за умовами та наслідками є симетричною до множини обмінів типу B^k .

Мета операцій обміну типу B^k : зменшення максимального з виступів за рахунок резерву пристрою з множини I_R з можливим перерозподілом резервів та виступів. У результаті обмінів цього типу змінюється склад множин I_R та I_Δ . На

відміну від обмінів типу A^k та B^k , в обміні цього типу робота з множини $J_s(\sigma)$ може не брати участь в обміні.

Умови виконання обміну типу B^k :

$$\theta > 0,$$

$$\theta \cdot k_h > \Delta_h(\sigma), \quad \theta \cdot k_s \geq R_s(\sigma), \quad \Delta_h(\sigma) + R_s(\sigma) > \theta. \quad (3.7)$$

У результаті обміну отримуємо розклад σ^1 , у якого:

$$R_h(\sigma^1) = \theta \cdot k_h - \Delta_h(\sigma) > 0, \quad \Delta_s(\sigma^1) = \theta \cdot k_s - R_s(\sigma).$$

При цьому, у новому розкладі $h \in I_R(\sigma^1)$, $s \notin I_R(\sigma^1)$, $\Delta_h(\sigma) > \Delta_s(\sigma^1)$ (тобто, розклад σ^1 не гірший за σ).

Обмін підтипу $1-0B^k$: переставляється робота ij з пристрою $h \in I_\Delta(\sigma)$ на пристрій $s \in I_R(\sigma)$, обмін задовольняє наступним умовам:

$$p_{ij} > R_s(\sigma), \quad (3.8)$$

$$\theta = p_{ij} - 0 > 0, \quad p_{ij} \cdot k_h > \Delta_h(\sigma), \quad \Delta_h(\sigma) + R_s(\sigma) > \theta.$$

Нерівність (3.8) є строгою нерівністю на відміну від відповідної умови (3.7), оскільки ми працюємо тільки з розкладами із множини Ψ .

Для отриманого в результаті обміну розкладу σ^1 виконується:

$$R_h(\sigma^1) = p_{ij} \cdot k_h - \Delta_h(\sigma) > 0, \quad \Delta_s(\sigma^1) = p_{ij} \cdot k_s - R_s(\sigma) > 0.$$

Обмін підтипу $1-1B^k$: міняються місцями робота ij з пристрою $h \in I_\Delta(\sigma)$ та робота kl з пристрою $s \in I_R(\sigma)$, роботи ij та kl задовольняють наступним умовам: $\theta = p_{ij} - p_{kl} > 0$, $(p_{ij} - p_{kl}) \cdot k_h > \Delta_h(\sigma)$, $(p_{ij} - p_{kl}) \cdot k_s \geq R_s(\sigma)$.

Для отриманого в результаті обміну розкладу σ^1 виконується:

$$R_h(\sigma^1) = (p_{ij} - p_{kl}) \cdot k_h - \Delta_h(\sigma), \quad \Delta_s(\sigma^1) = (p_{ij} - p_{kl}) \cdot k_s - R_s(\sigma).$$

Характеристики обмінів типу B^k представлені в таблиці 3.2.

У таблиці 3.1 розглянуті можливі співвідношення між величинами Θ , Δ_h та R_s , а також вказані відповідні їм значення зміни часткової цільової функції (мінімізація максимального із виступів пристроїв h та s).

Таблиця 3.1 – Вплив перестановок на значення цільової функції

Умови	Тип обміну	Зменшення значення цільової функції на пристрої h
1	2	3
$\theta \cdot k_h < \Delta_h, \theta \cdot k_s \leq R_s$	A^k	$\theta \cdot k_h$
$\theta \cdot k_h < \Delta_h, \theta \cdot k_s \leq R_s$	A^k	Δ_h
$\theta \cdot k_h < \Delta_h, \theta \cdot k_s \leq R_s$ $\theta \leq \left\lfloor \frac{\Delta_h + R_s}{2} \right\rfloor$	A^k	$\theta \cdot k_h$
$\theta \leq \Delta_h, \theta > R_s,$ $\theta \leq \left\lfloor \frac{\Delta_h + R_s}{2} \right\rfloor$	B^k	$\theta \cdot k_h$
$\theta \leq \Delta_h, \theta > R_s,$ $\theta \geq \left\lfloor \frac{\Delta_h + R_s}{2} \right\rfloor + 1$	B^k	$\Delta_h + R_s - \theta \cdot k_s$
$\theta > \Delta_h, \theta \geq R_s$	B^k	$\Delta_h + R_s - \theta \cdot k_s$

Узагальнимо ці результати: для обраної пари пристроїв $h-s$ у результаті операцій обміну типів A^k , B^k та B^k отримуємо таке зменшення значення максимального з виступів цих пристроїв у двох розкладах:

$$\Delta_h(\sigma) - \max\{\Delta_h(\sigma^1), \Delta_s(\sigma^1)\} = \min\{\Delta_h(\sigma), \theta \cdot k_h, \Delta_h(\sigma) + R_s(\sigma) - \theta \cdot k_s\}.$$

Коли вичерпані всі можливості покращення розкладу операціями обміну типами A^k , B^k та B^k пропонується використовувати операції обміну типу Γ^k , які зменшують значення $\max_i \Delta_i(\sigma)$ за рахунок перерозподілу виступів між пристроями.

Яскравою ілюстрацією необхідності розробки обмінів цього типу є розклад, представлений на рисунку 3.10.

Умови виконання обміну типу Γ^k :

$$\theta > 0, \theta \cdot k_h < \Delta_h(\sigma), \theta < \Delta_h(\sigma) - \Delta_s(\sigma).$$

У результаті обміну отримуємо розклад σ^1 , в якому:

$$\Delta_h(\sigma^1) = \Delta_h(\sigma) - \theta \cdot k_h, \quad \Delta_s(\sigma^1) = \Delta_s(\sigma) + \theta \cdot k_s.$$

Для розкладу σ^1 виконується: $\sum_{i=1}^m \Delta_i(\sigma^1) = \sum_{i=1}^m \Delta_i(\sigma)$,

але $\max\{\Delta_h(\sigma^1), \Delta_s(\sigma^1)\} < \max\{\Delta_h(\sigma), \Delta_s(\sigma)\}$, тобто, розклад σ^1 не гірший за σ .



Рисунок 3.10 – Приклад розкладу, який не можна покращити операціями обміну типами A^k , B^k та B^k

Обмін типу Γ^k . Міняються місцями робота ij з пристрою $h \in I_\Delta(\sigma)$ та робота kl з пристрою $s \in I_\Delta(\sigma) \cup I_0(\sigma)$ роботи ij та kl задовольняють наступним умовам:

$$\theta = p_{ij} - p_{kl} > 0,$$

$$(p_{ij} - p_{kl}) \cdot k_h < \Delta_h(\sigma),$$

$$(p_{ij} - p_{kl}) < \Delta_h(\sigma) - \Delta_s(\sigma).$$

В результаті обміну отримуємо розклад σ^1 , в якому:

$$\Delta_h(\sigma^1) = \Delta_h(\sigma) - (p_{ij} - p_{kl}) \cdot k_h,$$

$$\Delta_s(\sigma^1) = (p_{ij} - p_{kl}) \cdot k_s + \Delta_s(\sigma).$$

Приклад розкладу з обміном наведено у додатку А, рисунок А.4.

Для вибраної пари пристроїв $h - s$ у результаті обмінів типу Γ^k отримуємо таке зменшення значення максимального із виступів цих пристроїв у двох розкладах:

$$\begin{aligned} \Delta_h(\sigma) - \max\{\Delta_h(\sigma^1), \Delta_s(\sigma^1)\} &= \Delta_h(\sigma) - \max\{\Delta_h(\sigma) - \theta \cdot k_h, \Delta_s(\sigma) + \theta \cdot k_s\} = \\ &= \min\{\theta \cdot k_h; \Delta_h(\sigma) - \Delta_s(\sigma) - \theta \cdot k_s\}. \end{aligned}$$

Характеристики обмінів типу Γ^k представлені в таблиці 3.2.

Таблиця 3.2 – Властивості операцій обміну

Тип операції обміну	Умови, при яких виконується обмін	Характеристики результуючого розкладу σ^1
A^k	$\theta > 0$, $\theta k_h \leq \Delta_h(\sigma)$, $\theta k_s \leq R_s(\sigma)$	$\Delta_h(\sigma^1) = \Delta_h(\sigma) - \theta k_h$, $R_s(\sigma^1) = R_s(\sigma) - \theta k_s$
B^k	$\theta > 0$, $\theta k_s \leq \Delta_h(\sigma)$, $\theta k_h > R_s(\sigma)$	$\Delta_h(\sigma^1) = \Delta_h(\sigma) - \theta k_s$, $\Delta_s(\sigma^1) = \theta k_h - R_s(\sigma)$
B^k	$\theta > 0$, $\theta k_s > \Delta_h(\sigma)$, $\theta k_h \geq R_s(\sigma)$	$R_h(\sigma^1) = \theta k_s - \Delta_h(\sigma)$, $\Delta_s(\sigma^1) = \theta k_h - R_s(\sigma)$
Γ^k	$\theta > 0$, $\theta k_s < \Delta_h(\sigma)$, $\theta k_h < \Delta_h(\sigma) - \Delta_s(\sigma)$	$\Delta_h(\sigma^1) = \Delta_h(\sigma) - \theta k_s$, $\Delta_s(\sigma^1) = \Delta_s(\sigma) + \theta k_h$

Ці операцій обміну покладені в основу розробленої поліноміальної складової ПДС-алгоритму розв'язання задачі.

3.1.3.2.3 ПДС-алгоритм розв'язання задачі

Тепер на основі ДУО та розробленої множини операцій обміну побудований алгоритм визначення мінімального загального часу виконання робіт з відношенням передування паралельними пристроями. В основі даного методу лежить метод наведений у роботі [111].

Схема алгоритму

КРОК 0 Визначити оптимальний контур (достатні умови оптимальності).

КРОК 1 Побудувати початковий розклад σ^0 .

КРОК 2 $\sigma = \sigma^0$.

КРОК 3 Перевірити виконання достатніх умов оптимальності

ЯКЩО виконуються ДУО

ТО перейти на **КРОК 6** (σ – оптимальний розклад).

КРОК 4 Визначити пристрій h , якому відповідає максимальне значення виступу:

$$\Delta_h(\sigma) = \max_{v \in I_Z(\sigma)} \Delta_v(\sigma).$$

КРОК 5 Виконати для пристрою h , перебираючи усі пристрої

$s \in I_{\Delta}(\sigma) \cup I_0(\sigma) \cup I_R(\sigma)$, операцію обміну типів A^k , B^k , B^k або Γ^k .

ЯКЩО таких обмінів не знайшлось,

ТО перейти на **КРОК 6**,

ІНАКШЕ перейти на **КРОК 3**.

КРОК 6 Визначити максимальний загальний час виконання робіт:

$$C_v(\sigma) = C^* + \max_v \Delta_v, \quad v = \overline{1, m}.$$

КІНЕЦЬ АЛГОРИТМУ

Складність алгоритму складає $O(n^2W)$, де $W = \sum_{i=1}^n \sum_{j=1}^{n_i} p_{ij}$. Це пояснюється

наступним. На кожному кроці значення цільової функції (значення максимального із виступів) зменшується, щонайменше на 1. Тому, в найгіршому випадку алгоритм виконує кількість кроків, рівну значенню суми виступів у початковому розкладі σ^0 . До початку роботи алгоритму ця величина не відома. Величина W є дуже грубою верхньою оцінкою значення $\sum_{v=1}^m \Delta_v(\sigma^0)$. На кожному кроці трудомісткою операцією є пошук допустимих обмінів, яка зводиться до аналізу тривалостей всіх робіт двох обраних пристроїв (верхня межа кількості робіт на пристрої дорівнює n , звідси і n^2).

Якщо для результуючого розкладу σ виконується одна з ДУО, то цей розклад є оптимальним. В іншому випадку значення цільової функції розкладу σ відрізняється від її оптимального значення на величину не більше ніж $\max_v \Delta_v(\sigma)$. Тобто ця величина дає верхню межу відхилення значення критерія отриманого розв'язку від оптимального значення.

3.2 Опис експериментальних досліджень ефективності алгоритмів

3.2.1 Класифікація задач

Кількість одиниць часу, що витрачаються на досягнення оптимального розв'язку задачі алгоритмом та частка досягнення оптимуму визначає його ефективність. Ефективність алгоритму безпосередньо залежить від набору вхідних даних задачі. Неважко здогадатися, що ефективність алгоритму для систем, де на

кожному пристрої виконуються невелика кількість робіт, тривалість яких сильно різняться і для систем, де на кожному пристрої виконується велика кількість робіт, тривалість яких не сильно різняться одна від одної – буде різна.

Тож для проведення дослідження алгоритму та порівняти його ефективності за різних початкових умов, пропонується виділити наступні ознаки, за якими задачі можна класифікувати:

- 1) тривалість робіт;
- 2) кількість робіт;
- 3) кількість ланцюгів.

3.2.2 Результати експериментів

При проведенні експериментів для кожного класу задач було виділено два випадки А і Б.

У випадку А, при проведенні експериментів змінюється «ступінь розсіювання» тривалостей робіт. «Ступінь розсіювання» («СР») означає наскільки великою буде розбіжність між значеннями відносно їх медіани. Якщо середнє значення тривалості роботи є \bar{p} , а тривалості робіт в одній системі можуть бути як близькими одна до одної (отже, і до \bar{p}), так і значно відрізнятись. Тоді за це відповідатиме такий параметр, як Q (визначається як функція від \bar{p}). Він характеризує, наскільки великою буде розбіжність тривалість робіт відносно їх середнього значення. У залежності від величини коефіцієнта розсіювання Q встановлюється значення стандартного відхилення: $\sigma = \bar{p}Q$. Тоді початкові дані «ступеня розсіювання» будуть належати до одної з категорій:

- 1) категорія S – системи із "малим" значенням «ступеня розсіювання» тривалостей робіт, у яких $Q=0,1$;
- 2) категорія M – системи із "середнім" значенням «ступеня розсіювання» тривалостей робіт, у яких $Q=0,2$;
- 3) категорія L – системи із "великим" значенням «ступеня розсіювання» тривалостей робіт, у яких $Q=0,4$.

У випадку Б, змінюється «ступінь розсіювання» коефіцієнтів продуктивності пристроїв. Початкові дані «ступеня розсіювання» будуть належати до одної з категорій:

- 1) категорія S – системи із "малим" значенням «ступеня розсіювання» коефіцієнтів продуктивності пристроїв, у яких $1 \leq k \leq 2$;
- 2) категорія M – системи із "середнім" значенням «ступеня розсіювання» коефіцієнтів продуктивності пристроїв, у яких $1 \leq k \leq 3$;
- 3) категорія L – системи із "великим" значенням «ступеня розсіювання» коефіцієнтів продуктивності пристроїв, у яких $1 \leq k \leq 5$.

3.2.2.1 Тривалість робіт

Початкові дані тривалостей робіт будуть належати до одної з категорій:

- 1) категорія S – системи із "малим" значенням тривалостей робіт, у яких $1 \leq p \leq 5$;
- 2) категорія M – системи із "середнім" значенням тривалостей робіт, у яких $5 \leq p \leq 10$;
- 3) категорія L – системи із "великим" значенням тривалостей робіт, у яких $45 \leq p \leq 50$.

Випадок А

При різних значеннях «ступеня розсіювання» тривалостей робіт середня частка відхилення від оптимуму наведена в таблиці 3.3а.

Таблиця 3.3а – Результати експериментів при збільшенні тривалостей робіт

Тривалість робіт	«Ступінь розсіювання» тривалостей робіт	Частка відхилення А1	Частка відхилення А2	Частка відхилення А3	Частка відхилення А4
S	S	0,06	0,7	0,01	0,79
M	S	0,07	0,95	0	0,99
L	S	0,22	0,91	0,01	0,91

Тривалість робіт	«Ступінь розсіювання» тривалостей робіт	Частка відхилення А1	Частка відхилення А2	Частка відхилення А3	Частка відхилення А4
<i>S</i>	<i>S</i>	0,06	0,7	0,01	0,79
<i>S</i>	<i>M</i>	0,15	1,09	0,01	1,14
<i>M</i>	<i>M</i>	0,13	1,09	0,02	0,96
<i>L</i>	<i>M</i>	0,07	1,28	0,01	1,02
<i>S</i>	<i>L</i>	0,01	1,2	0,02	0,99
<i>M</i>	<i>L</i>	0,06	1,14	0,01	1,14
<i>L</i>	<i>L</i>	0,06	1,03	0,02	0,92

Випадок Б

При різних значеннях «ступеня розсіювання» коефіцієнтів продуктивності пристроїв результати експериментів наведено в таблиці 3.3б.

Таблиця 3.3б – Результати експериментів при збільшенні тривалостей робіт

Тривалість робіт	«Ступінь розсіювання» коефіцієнтів продуктивності	Частка відхилення А1	Частка відхилення А2	Частка відхилення А3	Частка відхилення А4
<i>S</i>	<i>S</i>	0,01	0,77	0,01	0,74
<i>M</i>	<i>S</i>	0,08	1,07	0	0,68
<i>L</i>	<i>S</i>	0,07	0,74	0,01	0,85
<i>S</i>	<i>M</i>	0,06	0,7	0,01	0,79
<i>M</i>	<i>M</i>	0,07	0,95	0	0,99
<i>L</i>	<i>M</i>	0,22	0,91	0,01	0,91
<i>S</i>	<i>L</i>	0,09	1,29	0,03	1,11
<i>M</i>	<i>L</i>	0,14	1,15	0,02	1,15
<i>L</i>	<i>L</i>	0,16	1,06	0,03	0,84

3.2.2.2 Кількість робіт

При збільшенні кількості робіт у ланцюгах та фіксованій кількості ланцюгів маємо наступні результати.

Випадок А

При різних значеннях «ступеня розсіювання» тривалостей робіт середня частка відхилення від оптимуму наведена в таблиці 3.4а.

Таблиця 3.4а – Результати експериментів при збільшенні кількості робіт

Кількість робіт	«Ступінь розсіювання» тривалостей робіт	Частка відхилення А1	Частка відхилення А2	Частка відхилення А3	Частка відхилення А4
<i>S</i>	<i>S</i>	0,05	0,11	0,03	0,03
<i>M</i>	<i>S</i>	0,04	0,25	0,02	0,21
<i>L</i>	<i>S</i>	0,16	0,78	0,02	0,68
<i>S</i>	<i>M</i>	0,04	0,14	0,03	0,05
<i>M</i>	<i>M</i>	0,02	0,52	0,01	0,28
<i>L</i>	<i>M</i>	0,12	0,93	0,01	1,03
<i>S</i>	<i>L</i>	0,04	0,06	0,02	0,02
<i>M</i>	<i>L</i>	0,02	0,27	0,01	0,45
<i>L</i>	<i>L</i>	0,03	1,18	0,04	1,22

Випадок Б

При різних значеннях «ступеня розсіювання» коефіцієнтів продуктивності пристроїв результати експериментів наведено в таблиці 3.4б.

Таблиця 3.4б – Результати експериментів при збільшенні кількості робіт

Кількість робіт	«Ступінь розсіювання» коефіцієнтів продуктивності	Частка відхилення А1	Частка відхилення А2	Частка відхилення А3	Частка відхилення А4
<i>S</i>	<i>S</i>	0,05	0,07	0,02	0,02
<i>M</i>	<i>S</i>	0,02	0,34	0,01	0,14
<i>L</i>	<i>S</i>	0,03	0,85	0	0,67
<i>S</i>	<i>M</i>	0,04	0,14	0,03	0,05
<i>M</i>	<i>M</i>	0,02	0,52	0,01	0,28
<i>L</i>	<i>M</i>	0,12	0,93	0,01	0,03

Продовження таблиці 3.4б

<i>S</i>	<i>L</i>	0,04	0,11	0,03	0,04
<i>M</i>	<i>L</i>	0,06	0,3	0,01	0,15
<i>L</i>	<i>L</i>	0,21	1,11	0,01	0,9

3.2.2.3 Кількість ланцюгів

При збільшенні кількості ланцюгів та фіксованій кількості робіт у ланцюгах маємо наступні результати.

Випадок А

При різних значеннях «ступеня розсіювання» тривалостей робіт середня частка відхилення від оптимуму наведена в таблиці 3.5а.

Таблиця 3.5а – Результати експериментів при збільшенні кількості ланцюгів

Кількість робіт	«Ступінь розсіювання» тривалостей робіт	Частка відхилення А1	Частка відхилення А2	Частка відхилення А3	Частка відхилення А4
<i>S</i>	<i>S</i>	0,16	0,24	0,05	0,14
<i>M</i>	<i>S</i>	0,05	0,31	0,05	0,13
<i>L</i>	<i>S</i>	0,02	0,11	0,01	0,11
<i>S</i>	<i>M</i>	0,11	0,4	0,07	0,39
<i>M</i>	<i>M</i>	0,03	0,21	0,01	0,1
<i>L</i>	<i>M</i>	0,01	0,23	0	0,18
<i>S</i>	<i>L</i>	0,1	0,36	0,03	0,3
<i>M</i>	<i>L</i>	0,05	0,19	0,01	0,22
<i>L</i>	<i>L</i>	0,01	0,18	0	0,31

Випадок Б

При різних значеннях «ступеня розсіювання» коефіцієнтів продуктивності пристроїв результати експериментів наведено в таблиці 3.5б.

Таблиця 3.56 – Результати експериментів при збільшенні кількості ланцюгів

Кількість робіт	«Ступінь розсіювання» коефіцієнтів продуктивності	Частка відхилення A1	Частка відхилення A2	Частка відхилення A3	Частка відхилення A4
<i>S</i>	<i>S</i>	0,06	0,35	0,07	0,13
<i>M</i>	<i>S</i>	0,04	0,22	0,01	0,08
<i>L</i>	<i>S</i>	0	0,13	0	0,04
<i>S</i>	<i>M</i>	0,11	0,4	0,07	0,39
<i>M</i>	<i>M</i>	0,03	0,21	0,01	0,1
<i>L</i>	<i>M</i>	0,01	0,23	0	0,18
<i>S</i>	<i>L</i>	0,13	0,25	0,07	0,19
<i>M</i>	<i>L</i>	0,02	0,34	0,01	0,15
<i>L</i>	<i>L</i>	0,01	0,11	0	0,23

У додатку В наведено графічна ілюстрація аналізу проведення результатів експериментів.

На підставі результатів експериментів отримані такі висновки: найкращі результати показали розрахунки з використанням початкового розкладу за алгоритмом A3 і найбільш критично на його значення впливає збільшення робіт у ланцюгах. Результат з використанням алгоритму A1 є близьким до A3 і також дає гарні результати. Також, зважаючи на те, що для знаходження початкового розкладу за алгоритмом A3 проводиться серія розрахунків і обирається кращий, то часу на розв'язок задачі за алгоритмом A1 є меншим.

Висновок до розділу 3

У розділі наведено результати дослідження задачі складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт.

Для задачі розроблено множину операцій обміну завданнями між пристроями, на основі достатніх умов оптимальності розроблено поліноміальну складову ПДС-

алгоритму вирішення задачі, що використовує цю множину операцій обміну між пристроями, які дозволяють послідовно покращувати значення критерію.

Для перевірки ефективності розроблених алгоритмів була проведена серія експериментів. Для цього запропонована класифікація індивідуальних задач складання розкладів, в основу якої покладені такі характеристики: кількість, середня тривалість робіт, «ступінь розсіювання» довжин робіт, «ступінь розсіювання» коефіцієнтів продуктивності пристроїв. Під час проведення експериментів на вхід алгоритму подавалися серії задач різних типів.

На підставі результатів експериментів отримані такі висновки: найкращі результати показали розрахунки з використанням початкового розкладу за алгоритмом А3 і найбільш критично на його значення впливає збільшення робіт у ланцюгах. Результат з використанням алгоритму А1 є близьким до А3 і також дає гарні результати. Також, зважаючи на те, що для знаходження початкового розкладу за алгоритмом А3 проводиться серія розрахунків і обирається кращий, то часу на розв'язок задачі за алгоритмом А1 є меншим.

4 ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ СИСТЕМИ

Для складання розкладу виконання робіт у ланцюгах з відношенням передування на паралельних пристроях розроблено програмний продукт.

Даний програмний продукт представляє собою web-застосування.

Архітектуру програмного продукту наведено на рисунку 4.1. Така архітектура дозволяє отримати наступні переваги:

- відокремлення логічних частин програми, що дозволяє їх окремо легко змінювати не коригуючи роботу тих частин, що не підлягали змінам;
- ізоляція сервера БД від клієнта шляхом розробки підсистеми уніфікованого доступу до даних, таким чином, забезпечується захист серверу від несанкціонованого доступу;
- розгортання програмного продукту на Windows Azure platform, що дозволяє не встановлювати його на комп'ютер користувача.

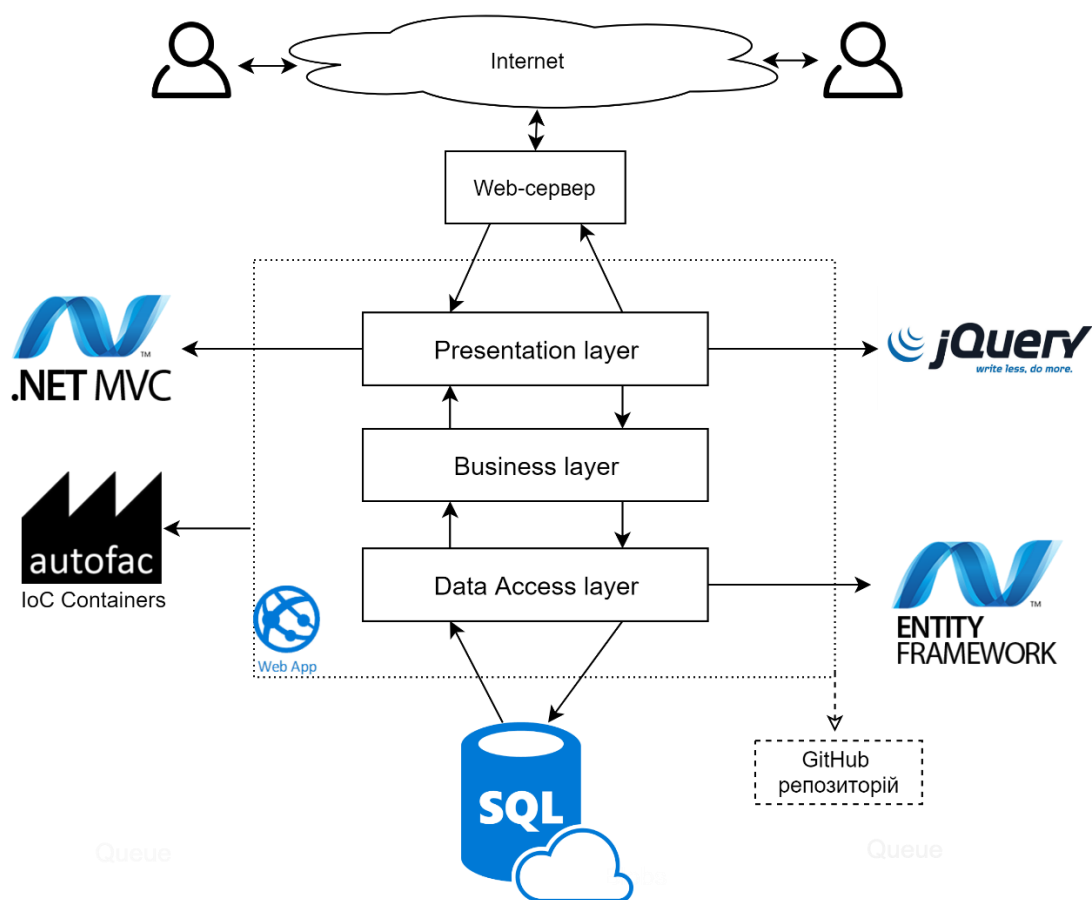


Рисунок 4.1 – Архітектура програмного продукту

Компонентами програмного продукту є:

- Web сервер. У роботі, у його якості, виступає основний компонент IIS (Internet Information Services) – веб-сервер, який дозволяє розміщувати в Інтернеті сайти. IIS – це набір серверів для декількох служб Інтернету від компанії Майкрософт. IIS підтримує різні протоколи передачі даних;

- сервер БД для роботи з даними Windows Azure SQL Databases, що базується на Microsoft SQL Server. Це хмарний сервіс від корпорації Microsoft, що надає можливість зберігання і обробки реляційних даних. Мова, що використовується для запитів – Transact-SQL. Перевагою такого збереження даних є стабільність та надійність збереження даних;

- репозиторій контролю версій GitHub для зберігання програмного коду. Github є одним із найбільших репозиторіїв для спільної розробки програмного забезпечення. Він дозволяє переглядати збережений код, має функцію підсвічування синтаксису для різних мов програмування. Github працює з системою контролю версій Git;

- сервер застосувань, в основу реалізації якого покладені розроблені в роботі методи, що описані в розділі 3 цієї роботи. Він складається з застосування складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм.

Розглянемо детальніше архітектуру сервера застосувань. Він представляє собою трирівневу архітектуру, наведену на рис. 4.1, яка складається з наступних рівнів:

- а) Рівень представлення даних. Це є рівень, з яким безпосередньо взаємодіє користувач. Цей рівень включає компоненти для користувача інтерфейсу, механізм отримання введення від користувача. Стосовно до ASP.Net MVC на даному рівні розташовані всі компоненти, які складають призначений для користувача інтерфейс (стилі, статичні сторінки html, javascript), а також моделі уявлень, контролери, об'єкти контексту запиту;

б) Бізнес рівень. Він містить набір компонентів, які відповідають за обробку отриманих від рівня уявлення даних, реалізує всю необхідну логіку застосунку, все обчислення, взаємодію з базою даних і передає рівню подання результати обробки;

с) Рівень доступу до даних. Даний рівень зберігає моделі, що описують суті, що використовуються, також тут розміщуються клас контексту даних Entity Framework. Тут також зберігаються репозиторії, через які рівень бізнес-логіки взаємодіє з базою даних.

Вся логіка рівнів була написана за допомогою мови C#.

4.1 Реалізація програмного продукту

4.1.1 Опис пакетів та класів програмного продукту

Класи програмного продукту було розділено на чотири пакети (рисунок 4.2) в залежності від функцій, які вони виконують:

- `algorithm.scheduling` – містить у собі класи, які реалізують алгоритми для знаходження ланцюгів та складання розкладу;
- `controllers` – містить класи-контролери, які обробляють запити, що надходять від клієнта;
- `data-access` – містить у собі класи, які здійснюють доступ до бази даних;
- `sql.entities` – містить у собі класи-сутності, які відображають реляційні дані з бази даних в об'єкти.

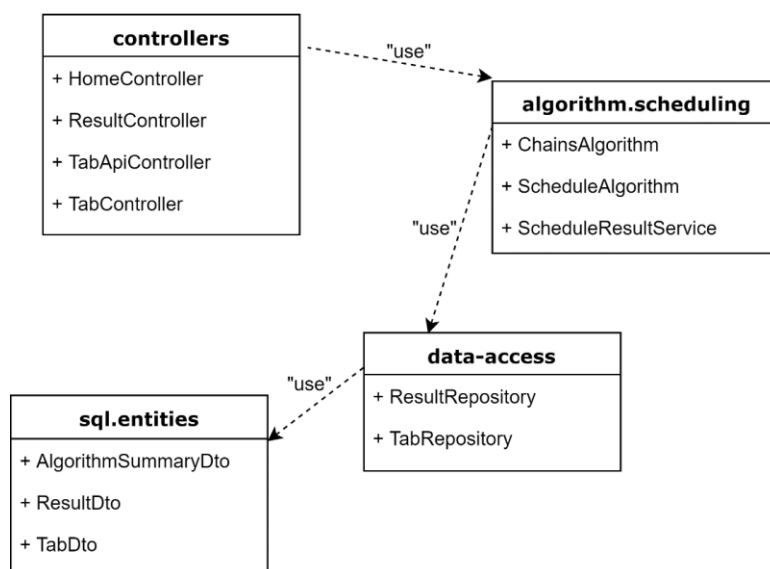


Рисунок 4.2 – Діаграма пакетів

4.1.2 Опис бази даних

В якості бази даних для програмного продукту використано реляційну базу даних Windows Azure SQL Databases, що базується на Microsoft SQL Server. Це хмарний сервіс від корпорації Microsoft, що надає можливість зберігання і обробки реляційних даних. Мова, що використовується для запитів – Transact-SQL. Перевагою такого збереження даних є стабільність та надійність збереження даних.

Структура схеми бази даних наведено на рисунку 4.3.

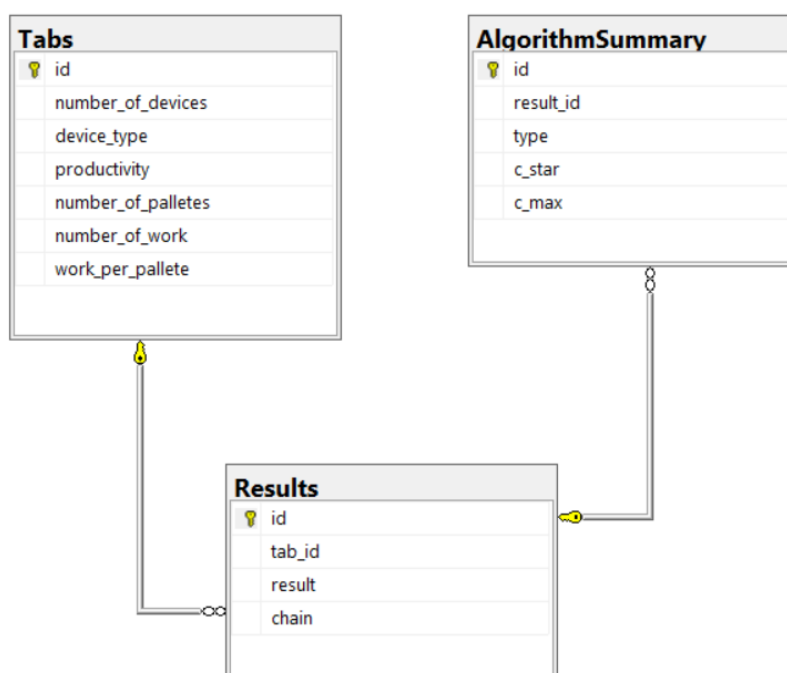


Рисунок 4.3 – Структура бази даних

4.2 Обґрунтування вибору технологій розробки

Програмний продукт розроблений з використанням наступних технологій:

- високорівнева мова програмування C#;
- реляційна база даних Windows Azure SQL Databases;
- ASP.NET MVC Framework;
- IoC-контейнер Autofac;
- Entity Framework.

В якості інтегрованого середовища розробки було використано Visual Studio 2017. Дане середовище має широкий спектр різних засобів інтеграції з іншими

засобами для розробки програмного забезпечення як системи контролю версій, web-сервери, покриття коду, фреймворки та ін [112].

Система контролю версій, яка була використана під час написання коду Git, є однією з найефективніших, надійних і високопродуктивних систем керування версіями, що надає гнучкі засоби нелінійної розробки, що базуються на відгалуженні та злитті гілок [113].

C# – об'єктно-орієнтована мова програмування з безпечною системою типізації для платформи .NET. Одна з найпотужніших мов, що швидко розвивається. C# є мовою з Cі-подібним синтаксисом і близький в цьому відношенні до C ++ і Java. Сформулюємо основні переваги даного мови [114]:

- об'єктно-орієнтована мова програмування;
- компонентно-орієнтований підхід до програмування;
- сприяє меншій машинно-архітектурній залежності результуючого програмного коду, гнучкості, переносимості і легкості повторного використання (фрагментів) програм;
- орієнтація на безпеку коду;
- уніфікована система типізації;
- розширена підтримка подієво-орієнтованого програмування.

Windows Azure SQL Databases – реляційна база даних, що базується на Microsoft SQL Server. Це хмарний сервіс від корпорації Microsoft, що надає можливість зберігання і обробки реляційних даних. Мова, що використовується для запитів – Transact-SQL. Перевагою такого збереження даних є стабільність та надійність збереження даних [115].

ASP.NET MVC Framework – фреймворк для створення веб-додатків, який реалізує шаблон Model-View-Controller. Платформа ASP.NET MVC базується на взаємодії трьох компонентів: контролера, моделі та представлення. Контролер приймає запити, обробляє введені дані користувача, взаємодіє з моделлю і представленням і повертає користувачу результат обробки запиту. Модель представляє рівень, що описує логіку організації даних у програмі. Представлення

отримує дані з контролера і генерує елементи призначеного для користувача інтерфейсу для відображення інформації. Перевага даного використання даного фреймворка в наступному [116]:

- більшість помилок видно на стадії розробки;
- призначені для користувача елементи управління дозволяють виділяти часто використовувані шаблони, такі як меню сайту;
- набір елементів управління і бібліотек класів дозволяє швидше розробляти додатки;
- можливість кешування даних, що використовуються на сторінці;
- можливість поділу візуальної частини та бізнес-логіки по різних файлах;
- розширена модель обробки запитів;
- наявність master-сторінок для завдання шаблонів оформлення сторінок;
- вбудована підтримка AJAX;
- має перевагу в швидкості в порівнянні з іншими технологіями, заснованими на скриптах.

ІоС-контейнер Autofac – фреймворк, який дозволяє використовувати архітектурне рішення інтеграції, яке спрощує розширення можливостей системи, при якому контроль над потоком управління програми залишається за каркасом [117].

.NET Entity Framework (EF) - об'єктно-орієнтована технологія доступу до даних, є object-relational mapping (ORM) рішенням для .NET Framework від Microsoft. Надає можливість взаємодії з об'єктами як за допомогою LINQ у вигляді LINQ to Entities, так і з використанням Entity SQL.

4.3 Огляд програмної реалізації продукту

Для складання розкладу робіт у рамках розробки програмного продукту були реалізовані наступні функції:

- ведення інформації про пристрої;
- ведення інформації про палету;
- складання розкладу виконання робіт:

- 1) формування ланцюгів;
- 2) формування розкладу;
- 3) графік відображення сформованого розкладу.

Розглянемо процес роботи програмного продукту від ролі користувача. Відкривши web-сторінку програмного продукту користувач потрапляє на головну сторінку (рисунк 4.4), де розташований його опис. Далі користувач має доступ до вкладок: «введення даних», «наявні дані», «розрахунки».



Рисунок 4.4 – Головна сторінка програмного продукту

На вкладці «Введення даних» (рисунк 4.5) користувач може ввести інформацію про пристрої та палету і після натиску на кнопку «розрахувати» відбувається формування розкладу виконання робіт, який можна побачити на вкладці «розрахунки». При цьому вхідні дані автоматично зберігаються на вкладці «наявні дані», що дає можливість надалі співставляти вхідні дані та результат розрахунків.

Пристрій	Продуктивність	Об'єкт	Час на виконання
1	3.4	1.1	9
2	1.8	1.2	7
		1.3	3
		1.4	7
		1.5	5
		1.6	5

Рисунок 4.5 – Вкладка «Введення даних»

На вкладці «Наявні дані» (рисунок 4.6) розміщені всі вхідні дані, введені користувачем для кожного створеного розкладу виконання робіт.

Складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт

Головна сторінка Введення даних **Наявні дані** Розрахунки

Дані Id:1101 Дані Id:1102 **Дані Id:1106**

Пристрої

Кількість: 2

Тип: Неідентичний

Пристрій	Продуктивність
1	1,0
2	3,0

Палета

Кількість рядів: 2

Кількість об'єктів у ряді: 7

Об'єкт	Час на виконання
1.1	5
1.2	8
1.3	9
1.4	5
1.5	7
1.6	6
1.7	8
2.1	9

Видалити

Результати

Рисунок 4.6 – Вкладка «Наявні дані»

На вкладці «Розрахунки» (рисунок 4.7) показано сформовані ланцюги, значення критерію, а також розміщено графіки відображення розкладу виконання робіт, що були отримані використовуючи початковий розклад за алгоритмами A1, A2, A3 та A4.

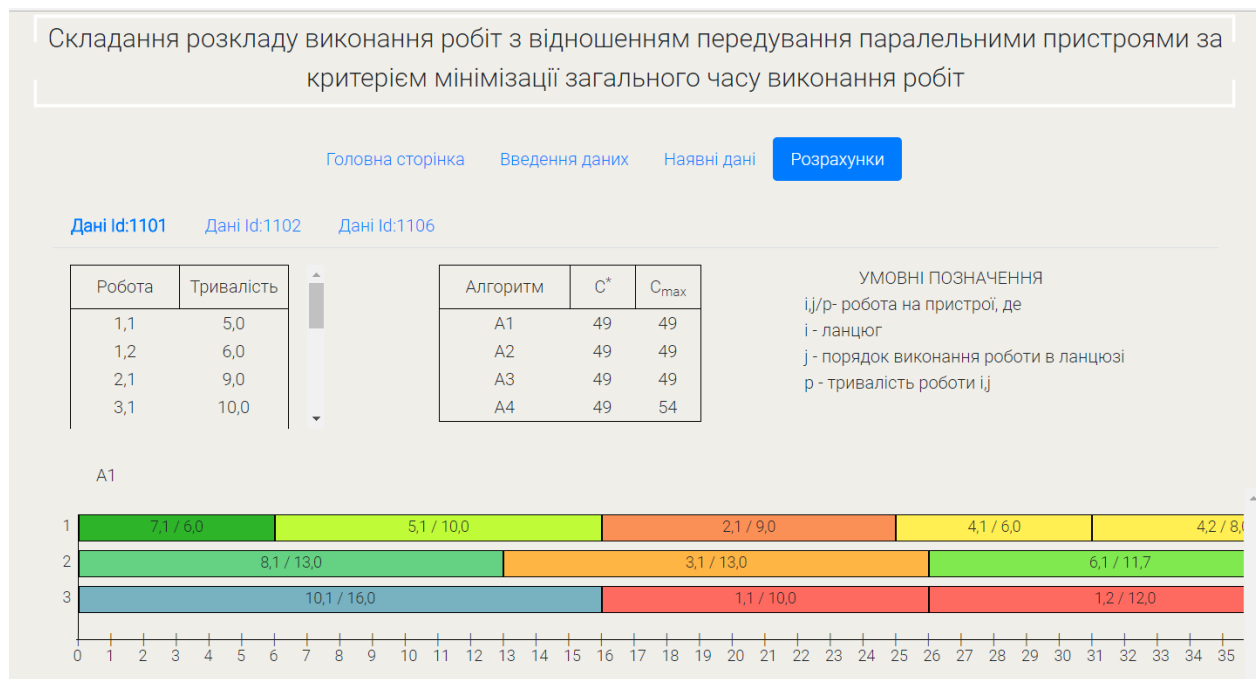


Рисунок 4.7 – Вкладка «Розрахунки»

Висновок до розділу 4

У розділі розглянуто розробку програмного продукту для складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт. Наведено опис розробленої архітектури програмного продукту, переваги якої було наведено. Розглянуті компоненти та принцип їх функціонування, а також архітектура самого сервера застосувань, яка представляє собою трирівневу архітектуру. Наведені пакети та класи програмного продукту з описом основних функцій, розроблена база даних з використанням реляційної бази даних Windows Azure SQL Databases. Наведено обґрунтування вибору технологій розробки, до яких увійшли наступні технології: високорівнева мова програмування C#, реляційна база даних Windows Azure SQL Databases, ASP.NET MVC Framework, Entity Framework, IoC-контейнер Autofac.

У рамках розробки програмного продукту було вирішено задачу складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт.

Наведено короткий огляд програмної реалізації програмного продукту та наведено основні частини інтерфейсу користувача.

ВИСНОВКИ

У *першому розділі* було розглянуто таку основну складову функціонування кожної компанії, як процес планування роботи виробництва та проблему планування, що виникає при побудові плану роботи підприємства. Було описано стан проблеми планування роботи підприємства на сьогоднішній день. Наведено сучасні методології управління виробництвом та оперативне управління основним виробництвом, основною складовою якого є оперативно-календарне планування та диспетчеризація, для яких і розробляється інформаційна технологія. Також було наведено різні системи управління виробництвом, як спосіб спростити управління виробничими процесами.

У *другому розділі* наведено існуючі на сьогоднішній день методи вирішення задач теорії розкладу та проведений аналіз застосування цих методів для реалізації схожих задач. У роботі наведено огляд результатів застосування розповсюджених методів розв'язання задач теорії розкладів. При проведенні аналізу літературних джерел було виявлено роботи, що пропонують різні підходи до розв'язання досліджуваної задач теорії розкладів. Також була описана змістовна постановка задачі. Та наведений приклад задачі для ідентичних пристроїв та пристроїв з однаковою продуктивністю.

У *третьому розділі* наведено результати дослідження задачі складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт.

Для задачі розроблено множину операцій обміну завданнями між пристроями, на основі достатніх умов оптимальності розроблено поліноміальну складову ПДС-алгоритму вирішення задачі, що використовує цю множину операцій обміну між пристроями, які дозволяють послідовно покращувати значення критерію.

Для перевірки ефективності розроблених алгоритмів була проведена серія експериментів. Для цього запропонована класифікація індивідуальних задач складання розкладів, в основу якої покладені такі характеристики: кількість, середня тривалість робіт, «ступінь розсіювання» довжин робіт, «ступінь розсіювання»

коефіцієнтів продуктивності пристроїв. Під час проведення експериментів на вхід алгоритму подавалися серії задач різних типів.

На підставі результатів експериментів отримані такі висновки: найкращі результати показали розрахунки з використанням початкового розкладу за алгоритмом A3 і найбільш критично на його значення впливає збільшення робіт у ланцюгах. Результат з використанням алгоритму A1 є близьким до A3 і також дає гарні результати. Також, зважаючи на те, що для знаходження початкового розкладу за алгоритмом A3 проводиться серія розрахунків і обирається кращий, то часу на розв'язок задачі за алгоритмом A1 є меншим.

У *четвертому розділі* розглянуто розробку програмного продукту для складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт. Наведено опис розробленої архітектури програмного продукту, переваги якої було наведено. Розглянуті компоненти та принцип їх функціонування, а також архітектура самого сервера застосувань, яка представляє собою тривірневу архітектуру. Наведені пакети та класи програмного продукту з описом основних функцій, розроблена база даних з використанням реляційної бази даних Windows Azure SQL Databases. Наведено обґрунтування вибору технологій розробки.

У рамках розробки програмного продукту було вирішено задачу складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт.

Наведено короткий огляд програмної реалізації програмного продукту.

ПЕРЕЛІК ПОСИЛАНЬ

1. Жданова О.Г., Сперкач М.О., Дубок К.В. Задача складання розкладу робіт з відношенням передування паралельними пристроями / О.Г. Жданова, М.О. Сперкач, К.В. Дубок // Матеріали Міжнародної наукової конференції на тему «Глобальне конкурентне середовище: розвиток сучасних соціально-економічних систем». – м. Кишинів, Республіка Молдова: 21 квітня 2017 р. – С. 156-159. (International scientific conference The global competitive environment: development of modern social and economic systems, April 21, 2017. Chisinau, Republic of Moldova : Baltija Publishing. 176 pages.)
2. Жданова О.Г., Сперкач М.О., Дубок К.В. Задача складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт / О. Г. Жданова, М. О. Сперкач, К. В. Дубок. // Матеріали науково-практичної конференції «Інформатика та обчислювальна техніка-IOT-2018». – м. Київ.: НТУУ «КПІ ім. Ігоря Сікорського», 23-24 квітня 2018 р.
3. Види планування роботи з персоналом підприємства [Електронний ресурс] // Навчальні матеріали онлайн – Режим доступу до ресурсу:http://pidruchniki.com/12590605/menedzhment/vidi_planuvannya_roboti_personalom_pidpriyemstva.
4. Senthilkumar P. Literature Review of Single Machine Scheduling Problem with Uniform Parallel Machines / P. Senthilkumar, S. Narayanan., 2010. – 18 с. – (Intelligent Information Management).
5. Герасимчук В.Г. Стратегічне управління підприємством. Графічне моделювання: Навч. посіб. / В.Г. Герасимчук — К: КНЕУ, 2000. — 360 с.
6. Кузин Б.И. Организация и оперативно-календарное планирование машиностроительного производства в АСУП / Б.И. Кузин, В.А. Дуболазов. – Л.: Изд-во Ленингр. ун-та, 1978. – 240 с.
7. Мизюн В.А. Управление производственными системами и процессами [Електронне видання] / В.А. Мизюн. – Самара: Издательство СНЦ РАН, 2012. – 211

с. – Режим доступа до ресурсу: http://www.cfin.ru/management/manufact/manufacturing_sys-03.shtml .

8. Антонов А.Н. Основы современной организации производства / А.Н. Антонов, Л.С. Морозова. – М.: Дело и сервис, 2004. – 578 с.

9. Горнев В.Ф. Оперативное управление в ГПС / В.Ф. Горнев, В.В. Емельянов, М.В. Овсянников. – М.: Машиностроение, 1990. – 256 с.

10. Жукова И.С. Проблемы организации производства в условиях смены технологических укладов / И.С. Жукова // Теоретические основы и практика организации производства: Юбилейный сб. науч. трудов. — Воронеж : ВГТУ , 2010. – С. 13 - 20.

11. Ковалев В.В. Анализ деятельности предприятия: учеб. [Текст] / В.В. Ковалев, О.Н. Волкова. — М.: издательство «Проспект», 2004. – 424 с.

12. Мизюн В. А. Модель конкурентоспособного производства / В. А. Мизюн // Аудит и финансовый анализ. – 2009. – № 5. – С. 314 – 344.

13. Мильнер Б.З. Теория организации. / Б.З. Мильнер. – 6-е изд. (перераб. и доп.). – М.: ИНФРА-М, 2008. – 797 с.

14. Мыльник В.В. Исследование систем управления / В.В. Мыльник, Б.П. Титаренко, В.А. Волочиенко. – 4-е издание. – М.: Академический проект, 2006. – 352 с.

15. Родионова В.Н. Понятие и механизм синхронизации производственных процессов / В.Н. Родионова // Организатор производства. – М.: Экономика и финансы, 2010. – № 3. – С.15–18.

16. Сергиенко И.В. Задачи дискретной оптимизации. Проблемы, методы решения, исследования / И.В. Сергиенко, В.П. Шило. – К.: Наукова думка, 2003. – 260 с.

17. Емельянов В.В. Теория и практика эволюционного моделирования / В.В. Емельянов, В. В. Курейчик. – М.: Физматлит, Наука, 2003. – 432 с.

18. Ho J. Makespan Minimization for m Parallel Identical Processors / J. Ho, J. Wong., 1995. – 14 с.

19. Метод комбинаторных эвристик для решения комбинаторных задач упорядочения и распределения ресурсов / Д.И. Батищев, Э.Д. Гудман, И.П. Норенков, М.Х. Прилуцкий. – М.: Информационные технологии, 1997. – С. 29-32.
20. Hundal T.S. An extension of Palmer's heuristic for the flow-shop scheduling problem / T.S. Hundal, J. Rajgopal // International Journal of Produktion Reasearch. – 1988. – №26ю – P. 1119 – 1124.
21. Gupta J.N.D. A functional heuristic algorithm for the flop-shop scheduling problem / J.N.D. Gupta // Operational Research Quartrrly. – 1971. – №2. – P. 39-47.
22. Cambell H.G. A heuristic algorithm for the n job, m machine sequencing problem / H.G. Cambell, R.A. Dudek, M.L. Smith // Management Science. – 1970 – №16. – P. 630-637.
23. Carlier J. Scheduling jobs with release dates and tails on identical machines to minimize the makespan. European Journal of Operational Research 1987; 29:298 –306.
24. Dannenbring D.G. A evolution of flow shop sequencing heuristics / D.G. Dannenbring // Management Scine. – 1977. – №23. – P. 1174-1182.
25. Ho J.C. A new heuristic algorithm for the n-job, M-machine problem / J.C. Ho, Y.-L. Chang // European Journal of Operational Research – 1991. – №52. – P. 194-202.
26. Ishibuchi H., Misaki S., Tanaka H. Modived simulated annealing algorithms for the flow shop sequencing problem / H. Ishibuchi, S. Misaki, H. Tanaka // European Journal of Operational Research. – 1995. – №81. – P. 388-398.
27. Domschke W. Produktionsplanung. Ablauforganisatorische Aspekte / Domschke W., Sholl A., Vob S. – Berlin.: Heildelberg, Springer Verlag, 2005. – 456 S.
28. Подчасова Т.П., Португал В.М. и др. Эвристические методы календарного планирования / Т.П.Подчасова, В.М. Португал и др. – Киев.: Техника, 1980. – 140 с.
29. Monma C.L. Analysis of heuristics for preemptive parallel machine scheduling with batch setup times [Текст] / C.L. Monma, C.N. Potts // Operations Research. – 1993. – Vol. 41. – P. 981–993.

30. Lee H. A hybrid bounding procedure for the workload allocation problem on parallel unrelated machines with setups [Текст] / H. Lee, M. Guignard // Journal of the Operational Research Society. – 1996. – Vol. 47. – P. 1247–1261.
31. Lawler E.L. Branch-and-bound methods: A Survey / E.L. Lawler, D.E. Wood // Oper. Res. – 1966. – Vol.4, §14. – P. 252 – 260.
32. Land A.H. An automatic method of solving discrete programming problems / A.H. Land, A.G. Doig // Econometrica. – 1960. – Vol.28. – P. 497–520.
33. Танаев В.С. Введение в теорию расписаний / В.С. Танаев, В.В.Шкурба. – М.: Наука, 1975. – 256 с.
34. Зак Ю.А. Прикладные задачи теории расписаний и маршрутизации перевозок / Ю.А. Зак. – М.: Книжный дом «ЛИБРОКОМ», 2012. – 394 с.
35. Жиглявский А.А. Методы поиска глобального экстремума / А.А. Жиглявский, А.Г. Жилинскас. – М.: Наука, 1991. – 205 с.
36. Гурин Л.С. Задачи и методы оптимального распределения ресурсов / Л.С. Гурин, Я.С. Дымарский, А.Д. Меркулов. – М.: Сов. Радио, 1968. – 464 с.
37. Зак Ю.А. Методы оптимизации и их применение в целлюлозно-бумажной промышленности / Ю.А. Зак, Р.М. Рейдман, А.А. Рувинский. – М.: Лесная промышленность, 1973. – 248 с.
38. Zack Yu. A. Methods of Multiextremal Optimization under Constraints for Separably Quasimonotone Functions / Yu Zack // Journal of Computer and Systems Sciences International, 2011. – vol.50, №3. – P. 37 – 391.
39. Glover F. Tabu Search, Part II / F. Glover // ORSA Journal on Computing, 1990. – vol.2, №1. – P. 4-32.
40. Курейчик В.М. Генетические алгоритмы: Монография / В.М. Курейчик. – Таганрог: Изд.ТРТУ, 1998. – 242 с.
41. Cleveland G.A. Using genetic algorithms to schedule flow shop releases / G.A. Cleveland Smith S.F.// In. Proceeding of theThird International Conference on Genetic Algorithms. Morgan Kaufmann Publishers. – San Mateo,California. – 1989. – P. 160-169.
42. Glover F. Tabu Search, Part I / F. Glover // ORSA Journal on Computing. – 1989 – Vol. 1, No 3. – P. 190-206.

43. Glover F. Tabu Search, Part II / F. Glover // ORSA Journal on Computing. – 1990 – Vol. 2, No 1. – P. 4-32.
44. Nissen Volker. Einfuhrung in Evolutionare Algorithmen. / Volker Nissen // Optimierung nach dem Vorbild der Evolution. – Vieweg, Munchen, 1997. –P. 345.
45. Goldberg David E. Genetic Algorithms in Search / David E. Goldberg // Optimization, and Machine Learning. – Adison-Wesley, 1998. – 403 s.
46. Michalewicz Z. Heuristic methods for evolutionary computation tech-niques / Z. Michalewicz //Journal of Heuristics. – 1995. - №1. – P. 177-206.
47. Michalewicz Z. Genetic Algorithms + Data Structures = Evolution Programs / Z. Michalewicz. – Springer, Berlin, 1999. – P. 67.
48. Domschke W. Produktionsplanung. Ablauforganisatorische Aspekte / Domschke W., Sholl A., Vob S. – Berlin.: Heildelberg, Springer Verlag, 2005. – 456 S.
49. Зак Ю.А. Определение порядка выполнения независимых операций на параллельных машинах / Ю.А. Зак // Изв. АН СССР. Техническая кибернетика. – К., 1969. – №2. – С. 15–20.
50. Корбут А.А. Дискретное программирование / А.А. Корбут, Ю.Ю. Финкельштейн. – М.: Наука, Физматгиз, 1969. – 368 с.
51. Krelle W. Ganzzahlige Programmierungen. Theorie und Anwendungen in der Praxis / W. Krelle // Unternehmenforschung, 1958. – №2. – S. 161 – 175.
52. Brucker P. Scheduling Algorithms / P. Brucker. – Springer-Verlag, Berlin, Heidelberg und New York, 2001. – P. 107 – 155.
53. Zimmermann H.-J. Netzplantechnik / H.-J. Zimmermann. – Verlag Walter de Gruyter, Berlin/New York, 1971. – S. 156.
54. Ore O. Theory of graphs, American mathematical Society / O. Ore. – 1962. – (Colloquium Publications). – Vol. 38. – 270 p.
55. Корбут А.А. Дискретное программирование / А.А. Корбут, Ю.Ю. Финкельштейн. – М.: Наука, Физматгиз, 1969. – 368 с.
56. Беллман Р. Динамическое программирование и современная теория управления = Dynamic Programming and Modern Control Theory / Р. Беллман, Р.Калаба. – Пер. с англ. – М.: Наука, 1969. – 119 с.

57. Лежнев А.В. Динамическое программирование в экономических задачах / А.В. Лежнев. – Бином. Лаборатория знаний, 2010. – 176 с.
58. Михалевич В.С. Вычислительные методы исследований и проектирования сложных систем / В.С. Михалевич, В.Л. Волкович. – М.: Фитматгизм, Наука, 1982. – 287 с.
59. Эванс Э. Предметно-ориентированное проектирование (DDD): структуризация сложных программных систем. : Пер. с англ. / Э.Эванс. – М.: ООО «И.Д. Вильямс», 2014. – 448 с.
60. Webster S. Dynamic programming algorithms for scheduling parallel machines with family setup times [Текст] / S. Webster, M. Azizoglu // Computers & Operations Research. – 2001. – Vol. 28. – P. 127–137.
61. Алгоритмы локального поиска и задачи оптимизации [Электронный ресурс] // Искусственный интеллект. Системы и модели – Режим доступа до ресурсу: <http://www.riai.org.ru/algoritmyi-lokalnogo-poiska-i-zadachi-optimizatsii.html>.
62. Alon N., Woeginger G.J., Yadid T. Approximation schemes for scheduling on parallel machines // J. of Scheduling.– 1998.– V. 1.– P. 55 – 66.
63. Mastrolilli M. Efficient Approximation Schemes for Scheduling Problems with Release Dates and Delivery Times // J. of Scheduling.– 2003.– V. 6, N 6.– P. 521 – 531.
64. Сперкач М.О. Задача складання розкладу виконання завдань паралельними приладами з метою мінімізації максимуму відхилення від директивного терміну моментів завершення приладами усіх завдань / О.А. Павлов, М.О. Сперкач, О.Г. Жданова // Математичне та комп'ютерне моделювання. Серія «Технічні науки». – Кам'янець-Подільський: Кам'янець-Подільський національний університет імені Івана Огієнка, 2014. – № 10. – с. 148 – 158.
65. E. Mokotoff. Parallel Machine Scheduling Problems: A Survey. Asia-Pacific Journal of Operational Research, 18(2):193-242, 2001.
66. T.C.E. Cheng. A State-of-the Art: A Review of the Parallel Machine Scheduling. European Journal of Operational Research, 18(2): 193-242, 2001
67. M.R. Garey, and D.S. Johnson. Computers and Intractability: A Guide to the Theory of NP-completeness, Freeman, San Francisco, 1979.

68. M.R. Garey, and D.S. Johnson. Strong NP-completeness Results: Motivation, Examples and Implications. *Journal of the ACM*, 25: 499-508, 1978.
69. Kramer A. A unified heuristic and an annotated bibliography for a large class of earliness-tardiness scheduling problems [Электронный ресурс]/ A. Kramer, A. Subramanian. – Brazil, Working Paper UFPB, 2015. – Режим доступа до ресурсу: [journals/corr/KramerS15](#).
70. Красовский Д.В. Алгоритмы решения минимаксной задачи составления расписания / Д.В. Красовский, М.Г. Фуругян // Известия РАН. Теория и системы управления. – 2008. – №5. – С. 69–74.
71. Гончаров Е.Н. Вероятностный поиск с запретами для дискретных задач безусловной оптимизации / Е.Н. Гончаров, Ю.А. Кочетов // Дискретный анализ и исследования операций. Сер.2. – 2002. – Т.9, №2. — С.13—30.
72. Кочетов Ю.А. Локальный поиск с чередующимися окрестностями / Ю. Кочетов, Н. Младенович, П. Хансен // Дискретный анализ и исследования операций. Серия 2. – 2003. – Т.10, №1. – С. 11–44.
73. Кочетов Ю.А. Использование чередующихся окрестностей для приближенного решения задачи календарного планирования с ограниченными ресурсами / Ю.А. Кочетов, А.А. Столяр // Дискретный анализ и исследования операций. Серия 2. – 2003. – Т.10, №2. – С. 29–56.
74. Алексеев О.Г. Комплексное применение методов дискретной оптимизации / О.Г. Алексеев. – М.: Наука, 1986.
75. Штовба С.Д. Муравьиные алгоритмы / С.Д. Штовба // ExponentaPro. Математика в приложениях. – 2003. – №4(4). – С. 70–75.
76. Lin Y. Unrelated Parallel Machines Scheduling Problem Using an Ant Colony Optimization Approach [Текст] / Y. Lin, H. Hsieh, F. Hsieh // World Academy of Science, Engineering & Technology – 2012.
77. Senthil Kumar. Ant Colony Approach for Makespan Minimization on Unrelated Parallel Machines [Электронный ресурс] / K.M. Senthil Kumar, V. Selladurai, K. Raja, K. Elangovan // International Journal of Engineering Science & Technology. – 2011. – Р. 3113 - 3120. – Режим доступа до ресурсу:

https://www.researchgate.net/publication/267717836_Ant_Colony_Approach_for_Makespan_Minimization_on_Unrelated_Parallel_Machines.

78. Glover F., Laguna M. Chapter 3: Tabu search/Ed. R. Colin Reeves. Modern Heuristics Techniques for Combinatorial Problems. Oxford: Blackwell Scientific Publications, 1993. P. 70-150.

79. Raghavan R. Probabilistic Contruction of Deterministic Algorithms: Approximating Packing Integer Programs / R. Raghavan // J. Computer and System Sciences. – 1988. – Vol. 37. – P. 130–143.

80. Костенко В.А. Синтез структур вычислительных систем реального времени с использованием генетических алгоритмов / В.А. Костенко, Р.Л. Смелянский, А.Г. Трекин // Программирование. – 2000. – №5 – С. 63–72.

81. Vacher J. P. Genetic algorithms in a multi-agent system. In Intelligence and Systems [Текст] / J. P. Vacher., T. Galinho, F. Lesage, A. Cardon // Proceedings, IEEE International Joint Symposia. – 1998. – P. 17–26.

82. Moon C. Integrated machine tool selection and operation sequencing with capacity and precedence constraints using genetic algorithm [Текст] / C. Moon, M. Lee, Y. Seo, Y.H. Lee // Computers & industrial engineering. – 2002. – 43(3). – P. 605–621.

83. A genetic algorithm approach for minimizing total tardiness in parallel machine scheduling problems [Текст] / Tufan Demiral, Nihan Cetin Demirel, Belgin Tasdelen, Vildan Ozkir // Proceedings of the World Congress on Engineering. – 2011. – P. 1190 - 1193.

84. Chaudhry Imran Ali. Minimizing flow time for the worker assignment problem in identical parallel machine models using GA [Текст] / Imran Ali Chaudhry // International Journal of Advanced Manufacturing Technology. – 2010. – P. 747 - 760.

85. Chen Jeng-Fung. Unrelated parallel-machine scheduling to minimize total weighted completion time [Электронный ресурс] / Jeng-Fung Chen // Springer-Verlag. – London, 2013. – P.8-25. – Режим доступа до ресурсу: https://www.infona.pl/resource/bwmeta1.element.springer-doi-10_1007-S10845-013-0842-Y.

86. Alcan Pelin. An Application with Non-identical Parallel Machines using Genetic Algorithm with the Help of Fuzzy Logic [Электронный ресурс] / Pelin Alcan //

Proceedings of the World Congress on Engineering. – 2011. – Режим доступа до ресурсу: http://www.iaeng.org/publication/WCE2011/WCE2011_pp1166-1169.pdf.

87. Алгоритмы: построение и анализ = Introduction to Algorithms / Кормен Т.Х. [та ін.]. – 2-е издание (пер. с англ.). – М.: Издательский дом «Вильямс», 2007. – 1296 с.

88. Vairam S. Parallel machine shop scheduling using memetic algorithm [Текст] / S. Vairam, V. Selladurai // Applied Mechanics and Materials, Vol. 573. – 2014. – P. 362-367.

89. Vairam S. Parallel machine shop scheduling using memetic algorithm [Текст] / S. Vairam, V. Selladurai // Applied Mechanics and Materials, Vol. 573. – 2014. – P. 362-367.

90. Guo Peng. Parallel machine scheduling with step deteriorating jobs and setup times by a hybrid discrete cuckoo search algorithm [Электронный ресурс] / Peng Guo, Wenming Cheng, Yi Wang // Engineering Optimization. – 2013. – P. 1-22. – Режим доступа до ресурсу: <http://arxiv.org/pdf/1309.1453v1.pdf>.

91. Parallel-Machine Scheduling Problem under the Job Rejection Constraint [Текст] / Weidong Li, Zhiban Chen, Xuejie Zhang, Jianping Li // Frontiers in Algorithmics. – China, 2014. – P.158-159.

92. Caniylmaz Erdal. An artificial bee colony algorithm approach for unrelated parallel machine scheduling with processing set restrictions, job sequence-dependent setup times, and due date [Текст] / Erdal Caniylmaz, Betul Benli, Mehmet Ilkay // International Journal of Advanced Manufacturing Technology. Springer-Verlag. – London, 2014. – P.9-12.

93. Rustogi Kabir A. Parallel Machine Scheduling: Impact of Adding Extra Machines [Текст] / Kabir Rustogi, Vitaly Strusevich // Operations Research. – 2013. – P. 1243 - 1257.

94. Zou Juan. Minimizing makespan with chain precedence constraints on identical parallel machines. [Текст] / Juan Zou, Yuzhong Zhang, Longchun Wang // Advances in Information Sciences & Service Sciences. Vol. 4, Issue 21. – 2012. – P. 1190 - 1193.

95. Cheng Zhenmin. An approximate algorithm for parallel machine scheduling problem to minimize total completion time [Текст] / Zhenmin Cheng // Computing & Information Systems. – 2010. – P. 187 - 198.
96. Kolahan F. A heuristic algorithm approach for scheduling of multi-criteria unrelated parallel machines [Текст] / F. Kolahan, V. Kayvanfar // World Academy of Science, Engineering & Technology – 2009. – P.102-106.
97. Laarhoven P. Job Shop Scheduling by Simulated Annealing / P. Laarhoven, E. Aarts, J. Lenstra // Operations Research. – 1992. – Vol. 40(1). – P. 113–125.
98. Shen C. Scheduling multiple job problems with guided evolutionary simulated annealing approach / C. Shen, Y. Pao, P. Yip // Proc. First IEEE Conf. on Evolutionary Computations. – Orlando, 1994. – P. 702–706.
99. Lin Shih-Wei. A multi-point simulated annealing heuristic for solving multiple objective unrelated parallel machine scheduling problems [Электронный ресурс] / Shih-Wei Lin, Kuo-Ching Ying // International Journal of Production Research. – 2014. – P. 1 -
Режим доступа до ресурсу:
https://www.researchgate.net/publication/269280769_A_multi-point_simulated_annealing_heuristic_for_solving_multiple_objective_unrelated_parallel_machine_scheduling_problems.
100. Zhang Rui. A simulated annealing-based heuristic algorithm for Job Shop scheduling to minimize lateness [Текст] / Rui Zhang // International Journal of Advanced Robotic Systems. – 2013. – P. 91-108.
101. Головкин Б.А. Расчет характеристик и планирования параллельных вычислительных процессов / Б.А. Головкин. – М: Радио и связь, 1983. –272 с.
102. Jeong Suk Jae. Parallel machine scheduling with earliness-tardiness penalties and space limits [Электронный ресурс] / Suk Jae Jeong, Kyung Sup Kim // Springer-Verlag. Volume 37, Issue 7. London, 2008. – P. 793-802. – Режим доступа до ресурсу: <http://link.springer.com/article/10.1007%2Fs00170-007-1027-7>.
103. Akyol Emine. A Variable Capacity Parallel Machine Scheduling Problem [Текст] / Emine Akyol, Tugba Sarac // Proceedings of the International Conference on Industrial Engine and Operations Management Istanbul, Turkey, 2012. – P. 548-554.

104. Unrelated parallel machines scheduling problem with sequence dependent setup times [Текст] / Vahid Kayvanfar, Amin Aalaei, Mahtab Hosseini, Mahdi Rajabi // Proceedings of the 2014 International Conference on Industrial Engineering and Operations Management. – 2014. – P. 1794 - 1803.

105. Toksari M.D. Minimizing the earliness/tardiness costs on parallel machine with learning effects and deteriorating jobs: a mixed nonlinear integer programming approach. [Текст] / M.D. Toksari, E. Guner // Springer-Verlag. – London, 2008. – P. 801-807.

106. Cheng Wenming. Variable Neighborhood Search for Parallel Machines Scheduling Problem with Step Deteriorating Jobs [Електронний ресурс] / Wenming Cheng, Peng Guo, Jian Liang // Mathematical Problems in Engineering. – 2012. – P. 1-11. – Режим доступу до ресурсу: <http://connection.ebscohost.com/c/articles/87029741/variable-neighborhood-search-parallel-machines-scheduling-problem-step-deteriorating-jobs>

107. Senthilkumar K.M. A Hybrid Algorithm Based on PSO and ACO Approach for Solving Combinatorial Fuzzy Unrelated Parallel Machine Scheduling Problem [Текст] / K.M. Senthilkumar, K. Raja // European Journal of Scientific Research. – 2011. – P. 87-104. – Режим доступу до ресурсу: <http://connection.ebscohost.com/c/articles/99708438/hybrid-clustering-algorithm-based-fuzzy-c-means-improved-particle-swarm-optimization>.

108. Park Y. Scheduling jobs on parallel machines applying neural network and heuristic rules [Текст] / Y. Park, S. Kim, Y.-H. Lee // Computers & Industrial Engineering. – 2000. – Vol. 38. – P. 189–202.

109. О. А. Павлов, О. Г. Жданова, М. О. Сперкач, задача складання розкладу виконання завдання паралельними приладами з метою мінімізації максимуму відхилення від директивного терміну моментів завершення приладами усіх завдань. — Київ: 2014.— 11 с.

110. М.З. Згуровський, А.А. Павлов, Е.Б. Мисюра ПДС-алгоритми і складно розв'язувальні задачі комбінаторної оптимізації, 2009.-18 с.

111. Сперкач М.О. Поліноміальна складова ПДС-алгоритму розв'язання однієї задачі теорії розкладів / О.А. Павлов, О.Г. Жданова, О.Б. Місюра, М.О. Сперкач // Технологический аудит и резервы производства, 2013. — №6/3 (14). — С.47-52.

112. Warren G. Visual Studio IDE overview [Електронний ресурс] / Genevieve Warren. — 2018. — Режим доступу до ресурсу: <https://docs.microsoft.com/en-us/visualstudio/ide/visual-studio-ide>.

113. Get Started with Git [Електронний ресурс]. — 2010. — Режим доступу до ресурсу: <https://alistapart.com/article/get-started-with-git>.

114. Introduction to the C# Language and the .NET Framework [Електронний ресурс]. — 2018. — Режим доступу до ресурсу: <https://docs.microsoft.com/en-us/dotnet/csharp/getting-started/introduction-to-the-csharp-language-and-the-net-framework>.

115. SQL Database The intelligent relational cloud database service [Електронний ресурс]. — 2018. — Режим доступу до ресурсу: <https://azure.microsoft.com/en-us/services/sql-database/>.

116. ASP.NET MVC Overview [Електронний ресурс]. — 2018. — Режим доступу до ресурсу: [https://msdn.microsoft.com/en-us/library/dd381412\(v=vs.108\).aspx](https://msdn.microsoft.com/en-us/library/dd381412(v=vs.108).aspx).

117. Blumhardt N. Dependency Injection with Autofac [Електронний ресурс] / Nicholas Blumhardt. — 2010. — Режим доступу до ресурсу: <https://www.codeproject.com/Articles/25380/Dependency-Injection-with-Autofac>.

ДОДАТКИ

ДОДАТОК А

**Ілюстрація перестановок різних типів між паралельними пристроями
для мінімізації максимального з виступів**

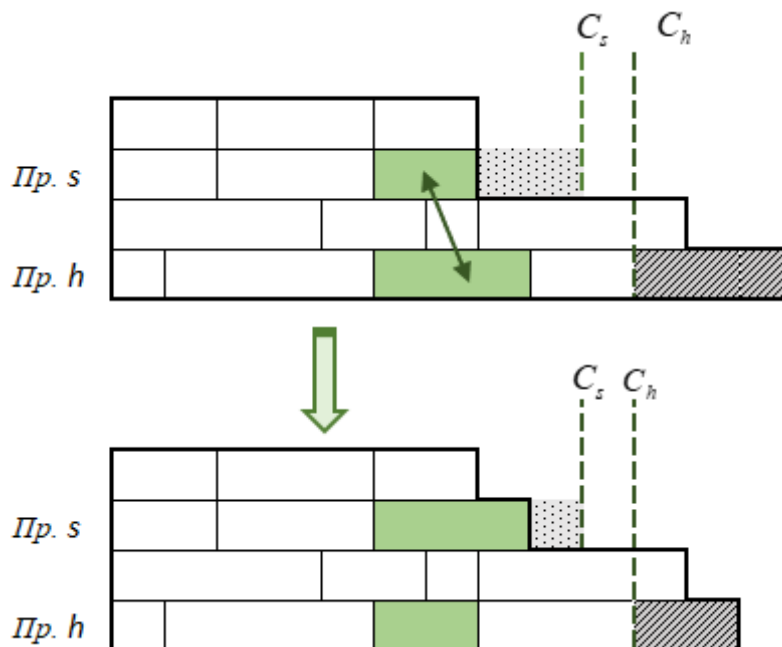


Рисунок А.1 – Приклад перестановки типу Ak

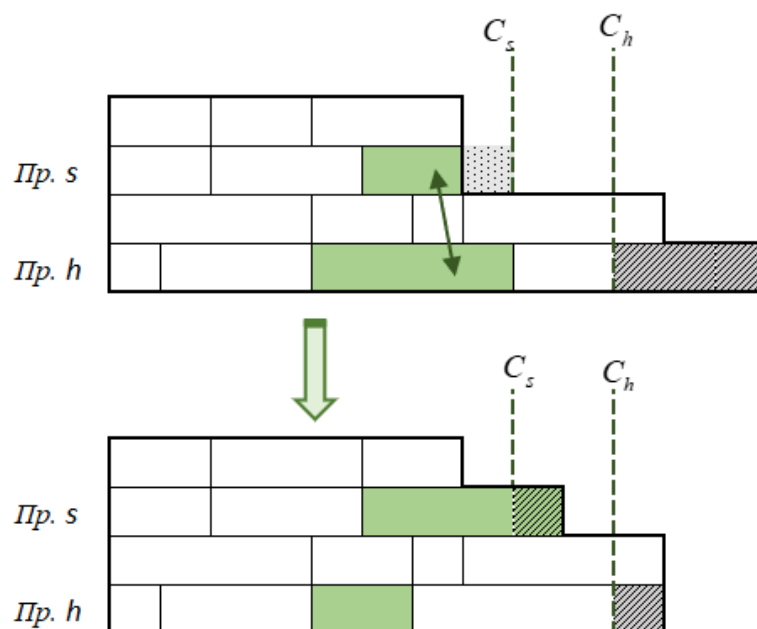
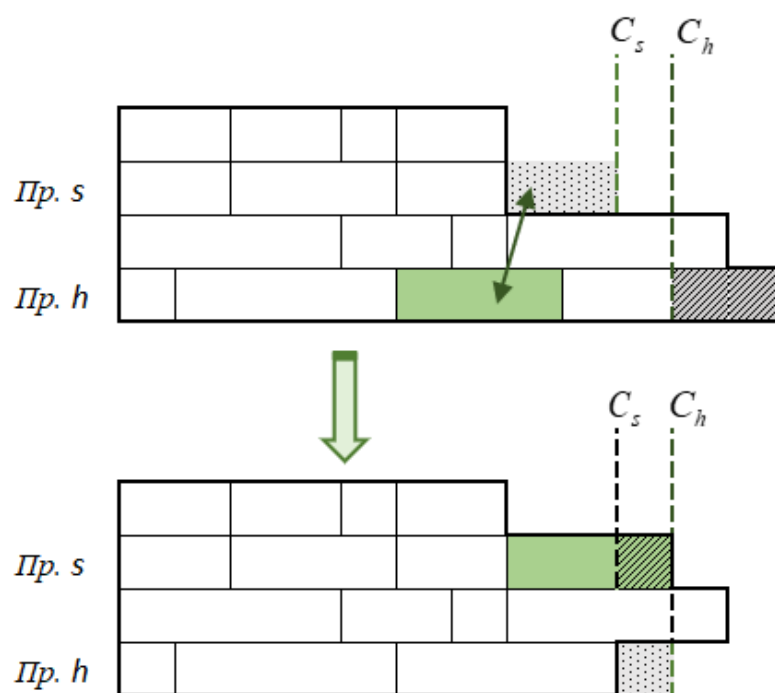
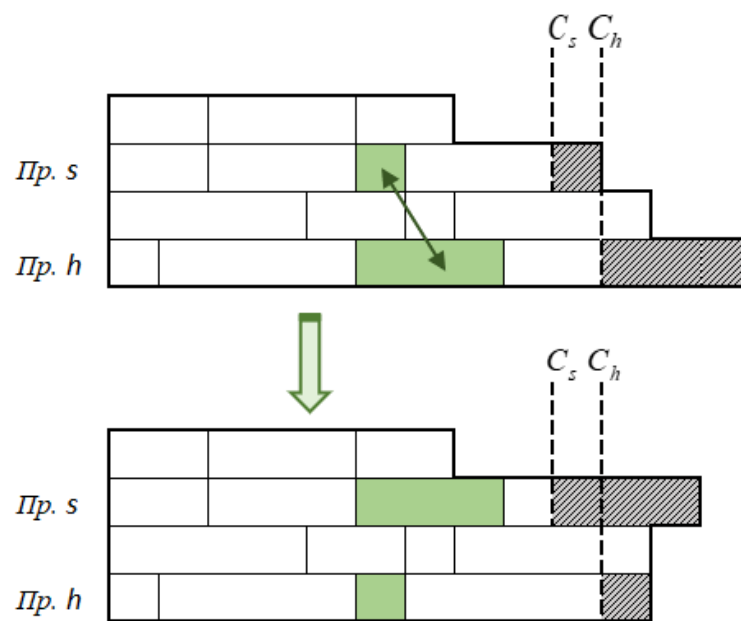


Рисунок А.2 – Приклад перестановки типу Bk

Рисунок А.3 – Пример перестановки типа B_k Рисунок А.4 – Пример перестановки типа Γ_k

ДОДАТОК Б

Приклад розв'язання задачі складання розкладу виконання робі з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт

Візьмемо для прикладу задачу, в якій кількість пристроїв $m = 3$. Коефіцієнти продуктивності пристроїв представлені у таблиці Б.1. Нехай палета складається з $S = 2$ рядів, кожен ряд має $H = 6$ комірок. Тривалості завантаження предметів у комірки представлені в таблиці Б.2.

Таблиця Б.1 – Коефіцієнти продуктивності пристроїв

Пристрій, i	1	2	3
Коеф. прод., k_i	1	1,5	2

Таблиця Б.2 – Тривалості завантаження предметів у комірки

Комірка, j	1	2	3	4	5	6
Тривалість, p_j	5	1	9	7	1	1
Комірка, j	7	8	9	10	11	12
Тривалість, p_j	6	4	3	1	5	3

На першому етапі розв'язку задачі сформуємо ланцюги робіт.

Визначимо точку поділу для першого ряду:

- a) перші три можливі точки поділу 2,3,4;
- b) найменша різниця між утвореними ланцюгами 6, коли точка поділу є 3;
- c) точка поділу 3 є центральна, тоді ланцюги першого ряду знайдені (таблиця Б.3).
- d) визначимо точку поділу для першого ряду:
- e) перші три можливі точки поділу 2,3,4;
- f) найменша різниця між утвореними ланцюгами 2, коли точка поділу є 2;
- g) точка поділу 3 є ліва – ставимо її як центральну;
- h) тепер можливі точки поділу 1,2,3;
- i) найменша різниця між утвореними ланцюгами 2, коли точка поділу є 2;

- ж) точка поділу 2 є центральна, тоді ланцюги другого ряду знайдені (таблиця Б.3).

Таблиця Б.3 – Ланцюги робіт

	Ланцюг, i			
Комірка, ij	1	2	3	
Тривалість, p_{ij}	9	1	5	
Комірка, ij	1	2	3	
Тривалість, p_{ij}	7	1	1	
Комірка, ij	1	2		
Тривалість, p_{ij}	4	6		
Комірка, ij	1	2	3	4
Тривалість, p_{ij}	3	1	5	3

Другий етап розв'язку задачі – це побудова розкладу за допомогою ПДС алгоритму.

Визначимо основні розрахункові величини алгоритму.

Мінімально можливий час, за який усі пристрої могли б виконати усі завдання:

$$C^* = \max \left\{ P_1 k_1; P_2 k_2; \dots; P_m k_m; \sum_{i=1}^n \sum_{j=1}^{n_i} \frac{1}{\sum_{v=1}^m \frac{1}{k_v p_{ij}}} \right\} \approx 21,23.$$

«Ідеальні» зведені тривалості зайнятості пристроїв наступні:

$$c_1^* = 21,23, \quad c_2^* = 14,15, \quad c_3^* = 10,62.$$

На першому кроці побудуємо початковий розклад алгоритмами **A1-A4**.

Алгоритм A1

Ітерація №1

- Сформуємо початкову множину $J^* := \{9, 7, 4, 3\}$.
- Оберемо роботу з найбільшою тривалістю 9.
- Визначимо множину пристроїв $M^* := \{1, 2, 3\}$.

- d) Обираємо пристрій 1 з M^* із максимальним невикористаним зведеним фондом робочого часу $f_v = 21, 23$.
- e) Призначаємо обрану роботу на пристрій.

Ітерація №2

- a) Сформуємо множину $J^* := \{1, 7, 4, 3\}$.
- b) Оберемо роботу з найбільшою тривалістю 7.
- c) Визначимо множину пристроїв $M^* := \{1, 2, 3\}$.
- d) Обираємо пристрій 2 з M^* із максимальним невикористаним зведеним фондом робочого часу $f_v = 14, 15$.
- e) Призначаємо обрану роботу на пристрій.

Ітерація №3

- a) Сформуємо множину $J^* := \{1, 1, 4, 3\}$.
- b) Оберемо роботу з найбільшою тривалістю 4.
- c) Визначимо множину пристроїв $M^* := \{1, 2, 3\}$.
- d) Обираємо пристрій 1 з M^* із максимальним невикористаним зведеним фондом робочого часу $f_v = 12, 23$.
- e) Призначаємо обрану роботу на пристрій.

Ітерація №4

- a) Сформуємо множину $J^* := \{1, 1, 6, 3\}$.
- b) Оберемо роботу з найбільшою тривалістю 6.
- c) Визначимо множину пристроїв $M^* := \{1\}$.
- d) Обираємо пристрій 1 з M^* із максимальним невикористаним зведеним фондом робочого часу $f_v = 8, 23$.
- e) Призначаємо обрану роботу на пристрій.

Ітерація №5

- a) Сформуємо множину $J^* := \{1, 1, 3\}$.
- b) Оберемо роботу з найбільшою тривалістю 3.

- c) Визначимо множину пристроїв $M^* := \{1, 2, 3\}$.
- d) Обираємо пристрій 3 з M^* із максимальним невикористаним зведеним фондом робочого часу $f_v = 10,62$.
- e) Призначаємо обрану роботу на пристрій.

Ітерація №6

- a) Сформуємо множину $J^* := \{1, 1, 1\}$.
- b) Оберемо роботу з найбільшою тривалістю 1.
- c) Визначимо множину пристроїв $M^* := \{1, 2\}$.
- d) Обираємо пристрій 2 з M^* із максимальним невикористаним зведеним фондом робочого часу $f_v = 7,15$.
- e) Призначаємо обрану роботу на пристрій.

Ітерація №7

- a) Сформуємо множину $J^* := \{5, 1, 1\}$.
- b) Оберемо роботу з найбільшою тривалістю 5.
- c) Визначимо множину пристроїв $M^* := \{1, 2\}$.
- d) Обираємо пристрій 2 з M^* із максимальним невикористаним зведеним фондом робочого часу $f_v = 6,15$.
- e) Призначаємо обрану роботу на пристрій.

Ітерація №8

- a) Сформуємо множину $J^* := \{1, 1\}$.
- b) Оберемо роботу з найбільшою тривалістю 1.
- c) Визначимо множину пристроїв $M^* := \{1, 2\}$.
- d) Обираємо пристрій 1 з M^* із максимальним невикористаним зведеним фондом робочого часу $f_v = 2,23$.
- e) Призначаємо обрану роботу на пристрій.

Ітерація №9

- a) Сформуємо множину $J^* := \{1, 1\}$.

- b) Оберемо роботу з найбільшою тривалістю 1.
- c) Визначимо множину пристроїв $M^* := \{1, 2, 3\}$.
- d) Обираємо пристрій 3 з M^* із максимальним невикористаним зведеним фондом робочого часу $f_v = 7,62$.
- e) Призначаємо обрану роботу на пристрій.

Ітерація №10

- a) Сформуємо множину $J^* := \{1, 5\}$.
- b) Оберемо роботу з найбільшою тривалістю 5.
- c) Визначимо множину пристроїв $M^* := \{1, 2, 3\}$.
- d) Обираємо пристрій 3 з M^* із максимальним невикористаним зведеним фондом робочого часу $f_v = 6,61$.
- e) Призначаємо обрану роботу на пристрій.

Ітерація №11

- a) Сформуємо множину $J^* := \{1, 3\}$.
- b) Оберемо роботу з найбільшою тривалістю 3.
- c) Визначимо множину пристроїв $M^* := \{1, 2, 3\}$.
- d) Обираємо пристрій 3 з M^* із максимальним невикористаним зведеним фондом робочого часу $f_v = 1,61$.
- e) Призначаємо обрану роботу на пристрій.

Ітерація №12

- a) Сформуємо множину $J^* := \{1\}$.
- b) Оберемо роботу з найбільшою тривалістю 3.
- c) Визначимо множину пристроїв $M^* := \{1, 3\}$.
- d) Обираємо пристрій 1 з M^* із максимальним невикористаним зведеним фондом робочого часу $f_v = 1,23$.
- e) Призначаємо обрану роботу на пристрій.

Усі роботи були розподілені на пристрої. На рисунку Б.1 представлено початковий розклад виконання робіт за алгоритмом *А1*.

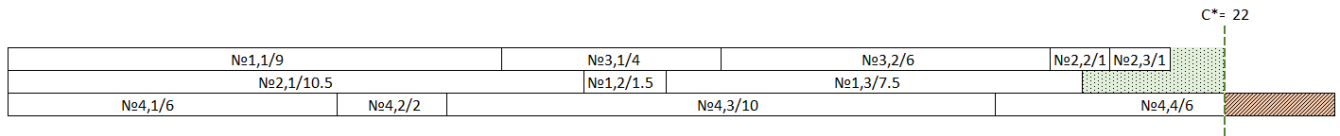


Рисунок Б.1 – Початковий розклад σ^0

Початковий розклад не є оптимальним, тож другим кроком є покращення результатів, який виконується за допомогою перестановок.

Ітерація №1

Визначимо множини $I_0(\sigma)$, $I_\Delta(\sigma)$ та $I_R(\sigma)$:

$$I_0(\sigma) \in \emptyset;$$

$$I_\Delta(\sigma) = \{3\}: \Delta_3(\sigma^0) = 2;$$

$$I_R(\sigma) = \{1, 2\}: R_1(\sigma^0) = 1, R_2(\sigma^0) = 2, 5.$$

Визначимо пристрої $h = I_\Delta(\sigma)$ та $s = I_R(\sigma)$: $h = 3$ – пристрій з максимальним значенням виступу. Визначимо роботу з пристроєм $s \in I_R(\sigma)$, з якою можлива операція обміну між h і s (обмін, який призводить до поліпшення розкладу та не порушує умову передування робіт). Для цього проведемо пошук по роботах ланцюгів доки не знайдемо роботу, для якої $\theta = \frac{P_{ij}}{k_1} - \frac{P_{kl}}{k_2} > 0$ і відношення передування у разі обміну для всіх робіт не порушиться.

Знайдені роботи для операції обміну наведено в таблиці Б.4.

Таблиця Б.4 – Роботи для операції обміну

№ пристрою	Коефіцієнт продуктивності	№ роботи	Тривалість роботи	θ	Δ	R
3	2	4,4	6	4	2	0
1	1	2,2	1		0	1

Після обміну робіт параметри для пари пристроїв 3-1 наведено в таблиці Б.5, де був проведений обмін типу B^k .

Таблиця Б.5 – Результат операції обміну

№ пристрою	Коефіцієнт продуктивності	№ роботи	Δ	R	Величина покращення	Тип обміну
3	2	4,4	0	2	1	B^k
1	1	2,2	1	0		

На рисунку Б.2 зображено розклад σ^1 , отриманий у результаті обміну.

№1,1/9										№3,1/4			№3,2/6					№4,4/3		№2,3/1
№2,1/10.5							№1,2/1.5			№1,3/7.5										
№4,1/6				№4,2/2		№4,3/10					№2,2/2									

Рисунок Б.2 – Розклад σ^1

Для розкладу σ^1 не виконується ДУО 2, продовжуємо роботу алгоритму.

Ітерація №2

Визначимо множини $I_0(\sigma)$, $I_\Delta(\sigma)$ та $I_R(\sigma)$:

$$I_0(\sigma) \in \emptyset;$$

$$I_\Delta(\sigma) = \{1\}: \Delta_1(\sigma^0) = 1;$$

$$I_R(\sigma) = \{2,3\}: R_2(\sigma^0) = 2,5, R_3(\sigma^0) = 2.$$

Визначимо пристрої $h = I_\Delta(\sigma)$ та $s = I_R(\sigma)$: $h=1$ – пристрій з максимальним значенням виступу. Визначимо роботу з пристроєм $s \in I_R(\sigma)$, з якою можлива операція обміну між h і s (обмін, який призводить до поліпшення розкладу та не порушує умову передування робіт). Для цього проведемо пошук по роботах ланцюгів допоки не знайдемо роботу, для якої $\theta = \frac{p_{ij}}{k_1} - \frac{p_{kl}}{k_2} > 0$ і відношення передування у разі обміну для всіх робіт не порушиться.

Знайдені роботи для операції обміну наведено в таблиці Б.6.

Таблиця Б.6 – Роботи для операції обміну

№ пристрою	Коефіцієнт продуктивності	№ роботи	Тривалість роботи	θ	Δ	R
1	1	2,3	6	1	2	0
2	1,5	-	-		0	2,5

Після обміну робіт параметри для пари пристроїв 3-1 наведено в таблиці Б.7, де був проведений обмін типу B^k .

Таблиця Б.7 – Результат операції обміну

№ пристрою	Коефіцієнт продуктивності	№ роботи	Δ	R	Величина покращення	Тип обміну
1	1	-	0	0	1	B^k
2	1,5	2,3	0	0		

На рисунку Б.3 зображено розклад σ^2 , отриманий у результаті обміну.

№1,1/9			№3,1/4		№3,2/6			№4,4/3	
№2,1/10.5				№1,2/1.5	№1,3/7.5				№2,3/1.5
№4,1/6		№4,2/2	№4,3/10				№2,2/2		

Рисунок Б.3 – Розклад σ^2

Отриманий розклад є оптимальним за ДУО. Визначимо для нього мінімальний час загального виконання робіт:

$$C_{\max}(\sigma) = C^* + \max_{1 \leq v \leq m} \Delta_v(\sigma) = 22 + 0 = 22.$$

ДОДАТОК В

Результати проведення експериметів

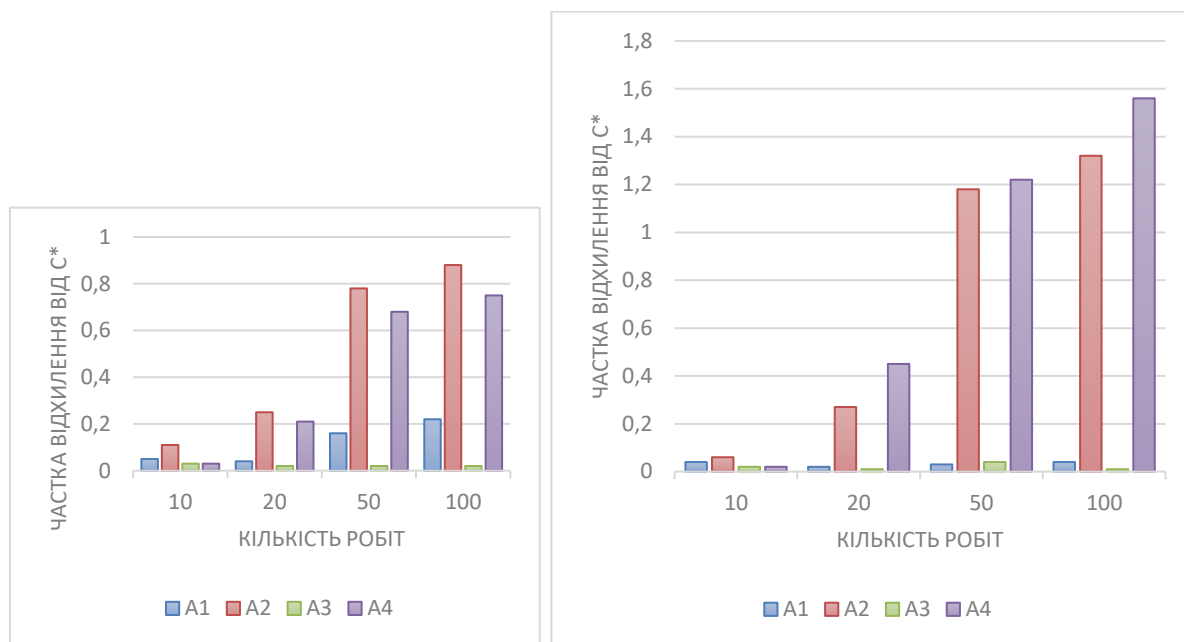


Рисунок В.1а – Кількість робіт (при різних значеннях «ступеня розсіювання» тривалостей робіт)

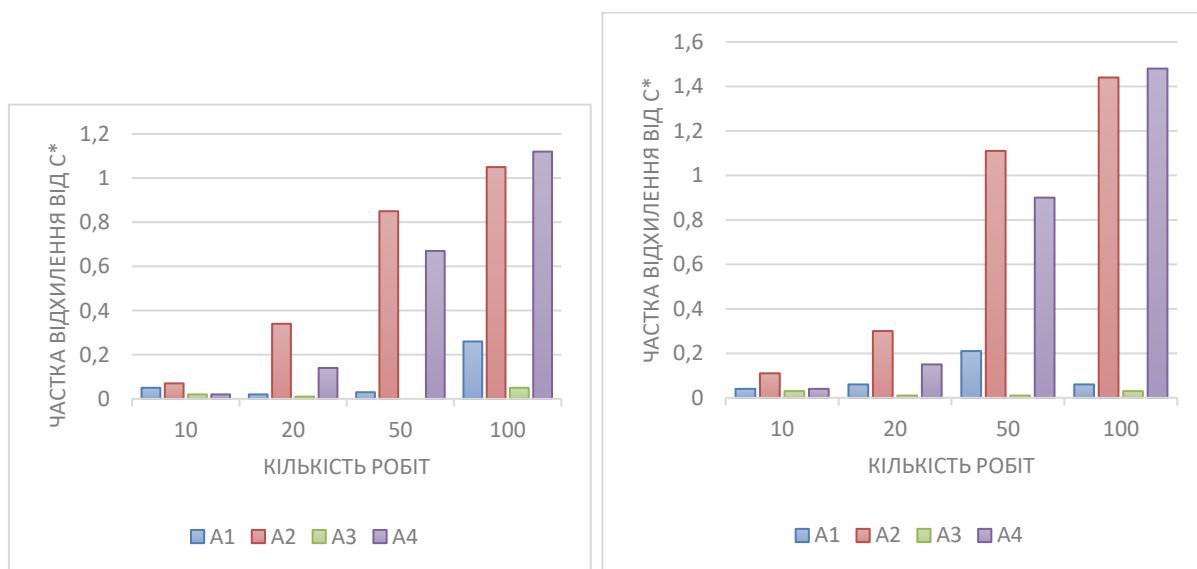


Рисунок В.1б – Кількість робіт (при різних значеннях «ступеня розсіювання» коефіцієнтів продуктивності пристроїв)

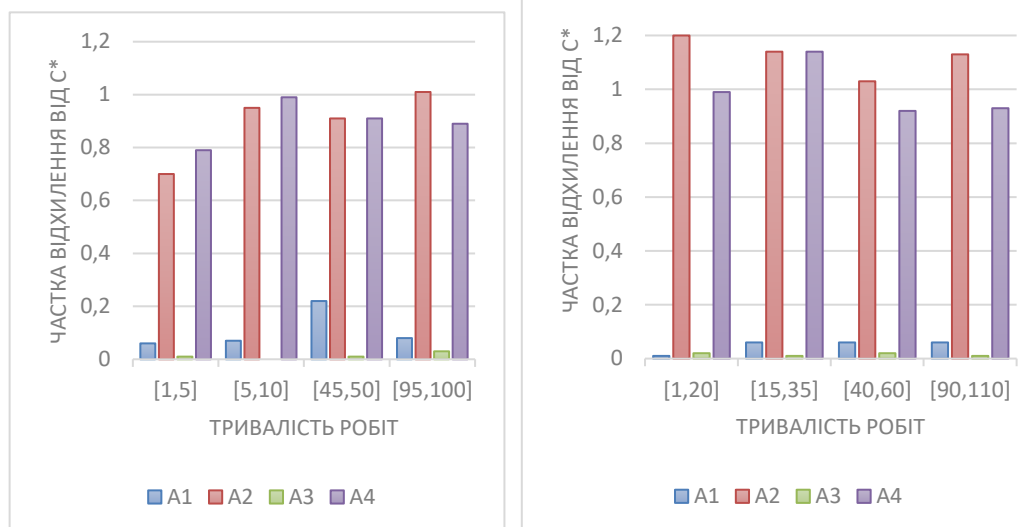


Рисунок В.2а – Тривалість робіт (при різних значеннях «ступеня розсіювання» тривалостей робіт)

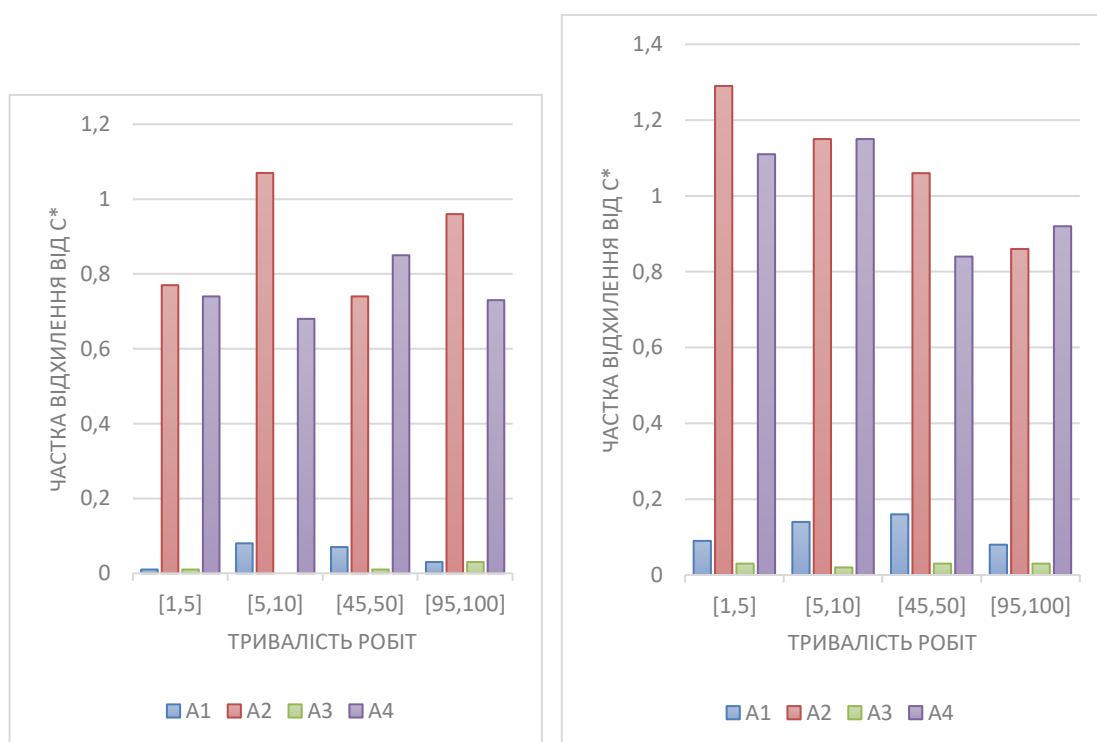


Рисунок В.2б – Тривалість робіт (при різних значеннях «ступеня розсіювання» коефіцієнтів продуктивності пристроїв)

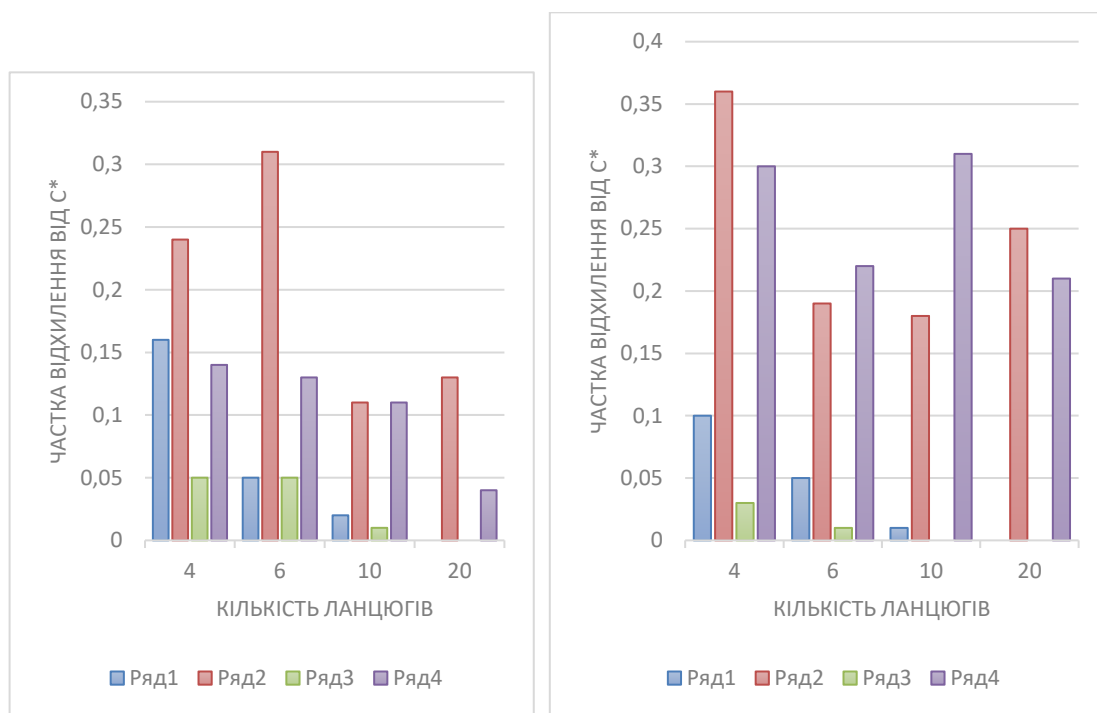


Рисунок В.3а – Кількість ланцюгів (при різних значеннях «ступеня розсіювання» тривалостей робіт)

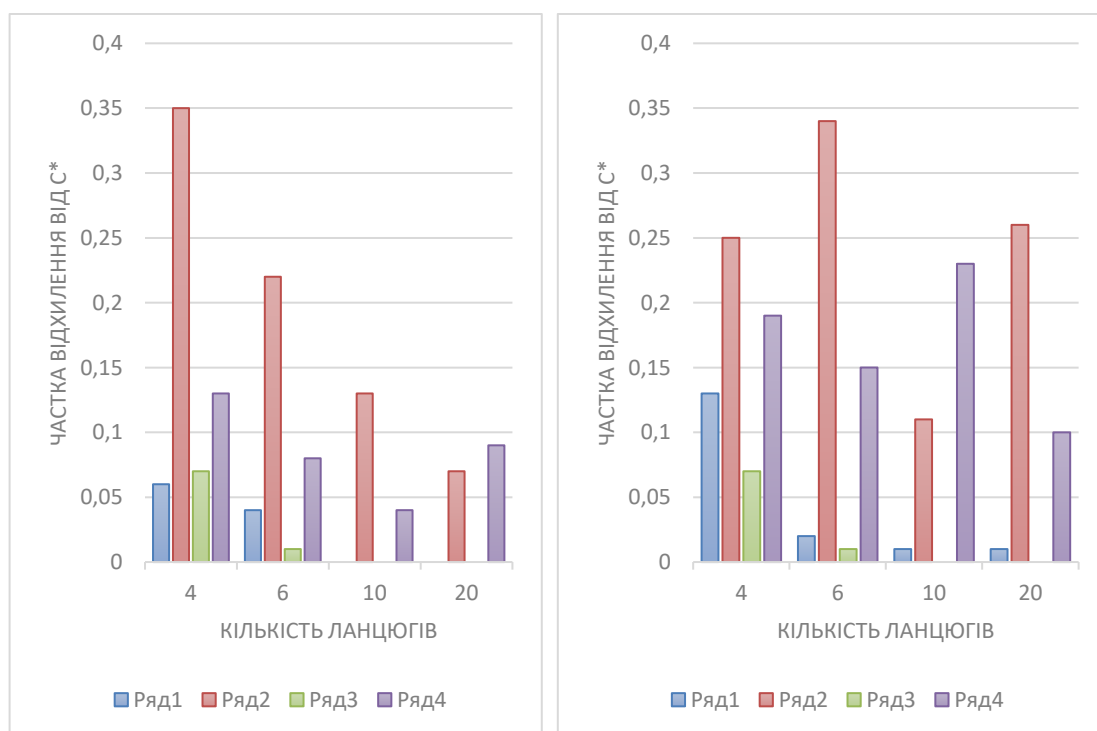


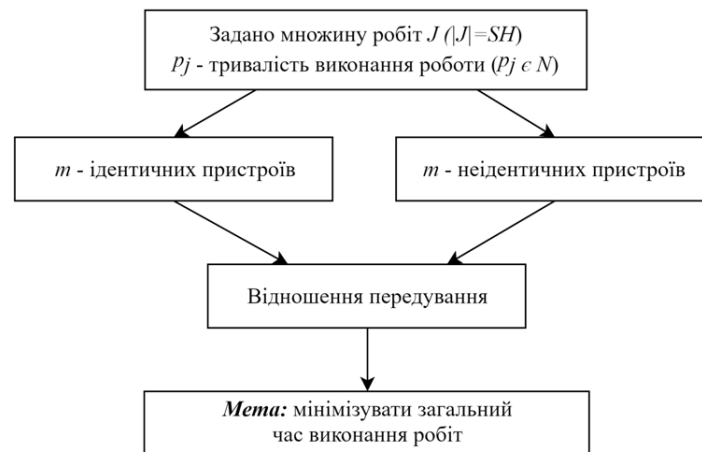
Рисунок В.3б – Кількість ланцюгів (при різних значеннях «ступеня розсіювання» коефіцієнтів продуктивності пристроїв)

ДОДАТОК Г
Графічний матеріал

ПЛАКАТ 1 Математична модель

МАТЕМАТИЧНА МОДЕЛЬ

Схематичне представлення задач дослідження



Достатні умови оптимальності розкладів

Випадок 1

$$C_{\max}(\sigma) = C^* = \sum_{i=1}^n \sum_{j=1}^{n_i} \frac{1}{\sum_{v=1}^m \frac{1}{k_v p_{ij}}}$$

Випадок 2

$$C_{\max}(\sigma) = C^* = \max \{P_1 k_1; P_2 k_2; \dots; P_{m-1} k_{m-1}\}$$

Нижня оцінка критерію C^*

$$C^* = \max \left\{ P_1 k_1; P_2 k_2; \dots; P_{m-1} k_{m-1}; \sum_{i=1}^n \sum_{j=1}^{n_i} \frac{1}{\sum_{v=1}^m \frac{1}{k_v p_{ij}}} \right\}$$

Основні позначення

 $J_v(\sigma)$ – упорядкована множина робіт, що виконується пристроєм v , $v = \overline{1, m}$; $C_v(\sigma) = \sum_{ij \in J_v(\sigma)} k_v p_{ij}$ – тривалість зайнятості пристрою v , $v = \overline{1, m}$; $\Delta_v(\sigma) = \max \{0; C_v(\sigma) - C^*\} = \max \left\{ 0; \sum_{ij \in J_v(\sigma)} k_v p_{ij} - C^* \right\}$ – виступ пристрою v , $v = \overline{1, m}$; $R_v(\sigma) = \max \{0; C^* - C_v(\sigma)\} = \max \left\{ 0; C^* - \sum_{ij \in J_v(\sigma)} k_v p_{ij} \right\}$ – резерв пристрою v , $v = \overline{1, m}$.

З урахуванням позначень критерій оптимізації вихідної задачі (мінімізація загального часу виконання робіт) має вигляд:

$$C_{\max}(\sigma) = C^* + \max_{1 \leq v \leq m} \Delta_v(\sigma) \rightarrow \min$$

Демонстраційний плакат до магістерської дисертації

на тему «Задача складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт»

Виконала студентка гр. ІС-61м

Дубок Катерина Вадентинівна

Керівник

Сперкач Майя Олегівна

ПЛАКАТ 2 Схема ПДС алгоритму

СХЕМА ПДС АЛГОРИТМУ

Схема алгоритму знаходження ланцюгів

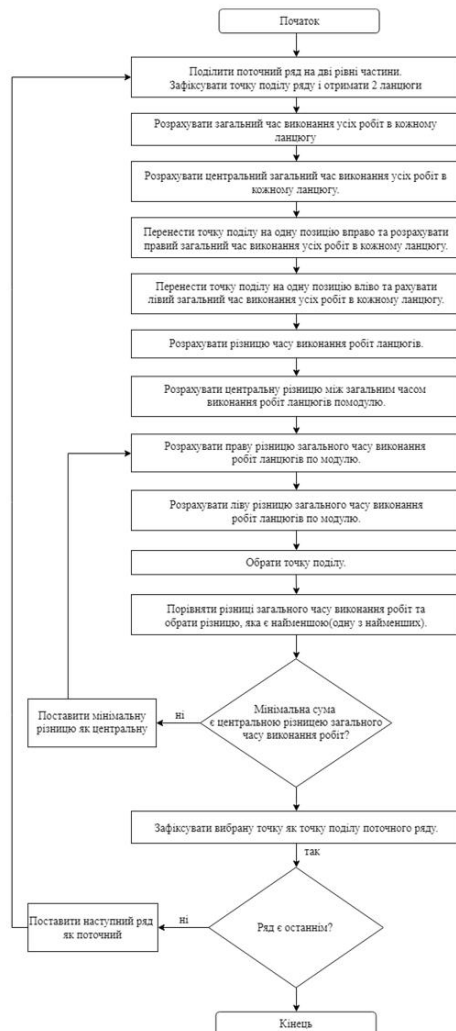
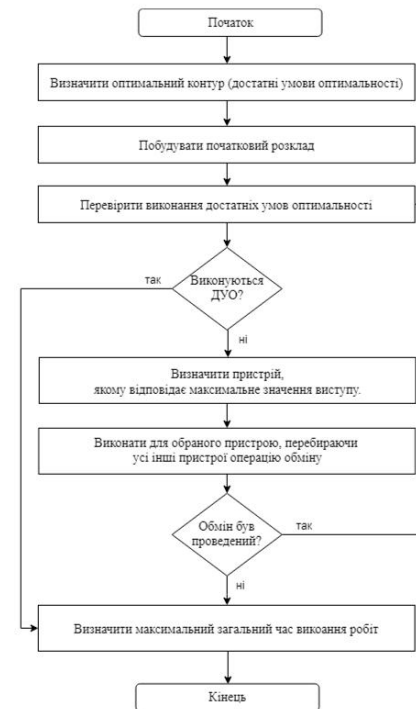


Схема ПДС алгоритму



Демонстраційний плакат до магістерської дисертації
на тему «Задача складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт»

Виконала студентка гр. ІС-61м

Керівник

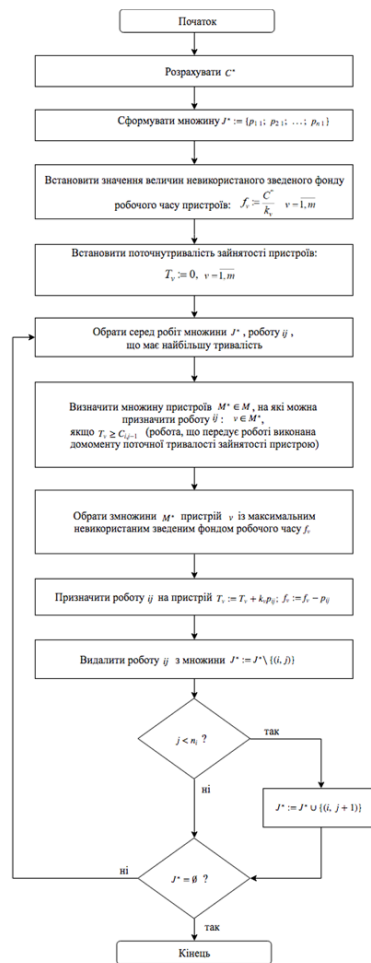
Дубок Катерина Вадентинівна

Сперкач Майя Олегівна

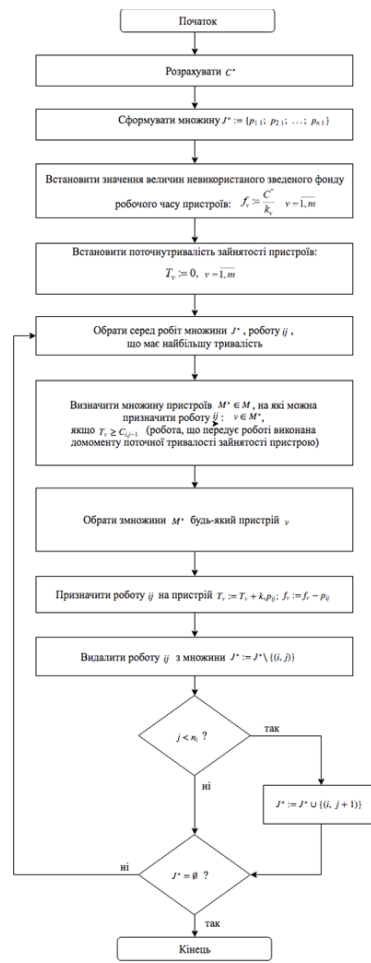
ПЛАКАТ 3 Схеми алгоритмів початкового розкладу A1-A4

СХЕМИ АЛГОРИТМІВ ПОЧАТКОВОГО РОЗКЛАДУ A1-A4

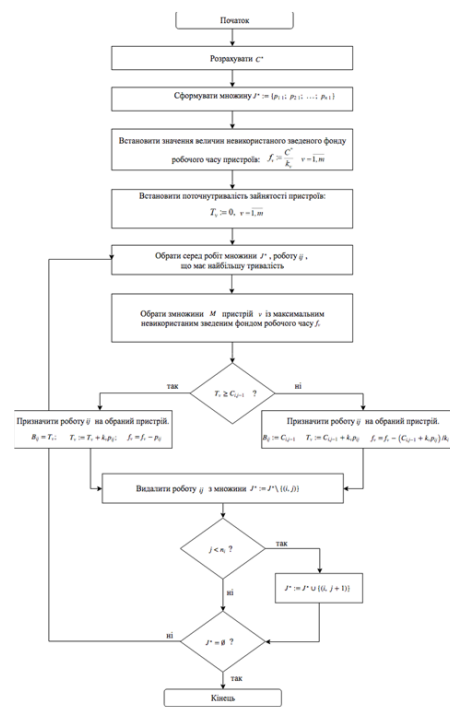
Алгоритм A1



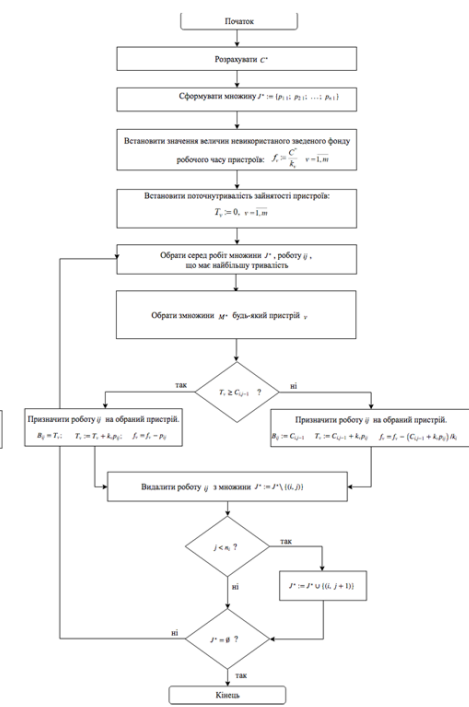
Алгоритм A3



Алгоритм A2



Алгоритм A4



Демонстраційний плакат до магістерської дисертації
на тему «Задача складання розкладу виконання робіт з відношенням передування паралельними
пристроями за критерієм мінімізації загального часу виконання робіт»

Виконала студентка гр. ІС-61м

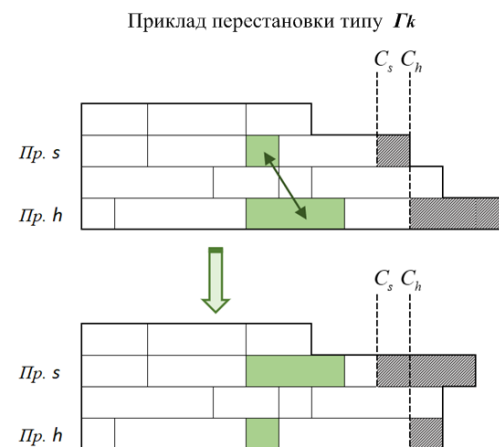
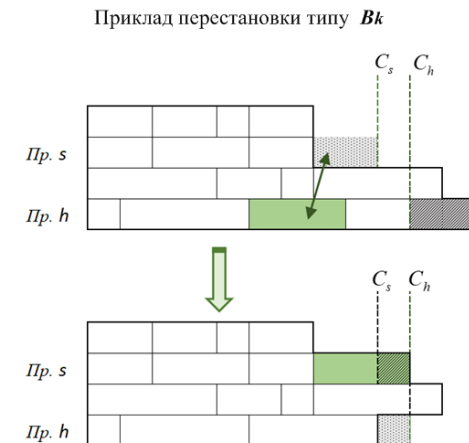
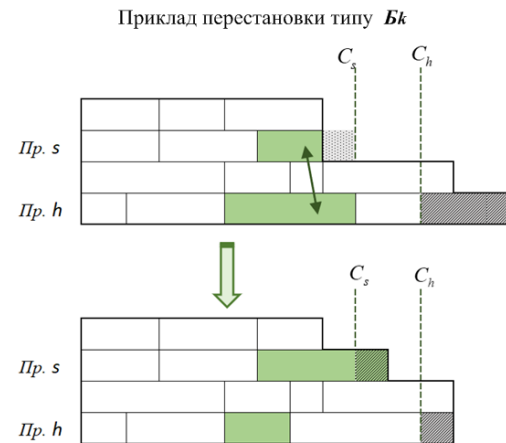
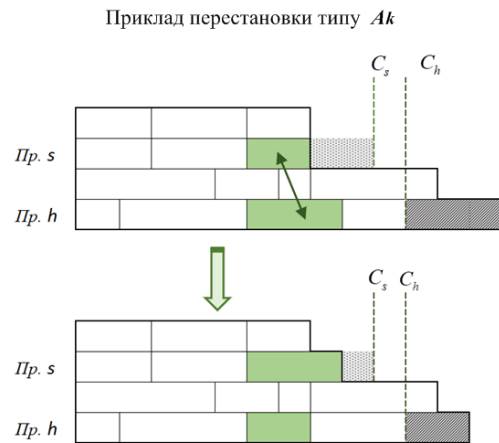
Керівник

Дубок Катерина Вадентинівна

Сперкач Майя Олегівна

ПЛАКАТ 4 Ілюстрація перестановок різних типів

ІЛЮСТРАЦІЯ ПЕРЕСТАНОВОК РІЗНИХ ТИПІВ



Демонстраційний плакат до магістерської дисертації
на тему «Задача складання розкладу виконання робіт з відношенням передування паралельними
пристроями за критерієм мінімізації загального часу виконання робіт»

Виконала студентка гр. ІС-61м

Дубок Катерина Вадентинівна

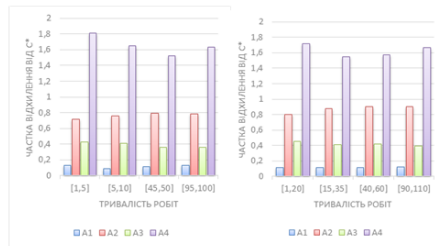
Керівник

Сперкач Майя Олегівна

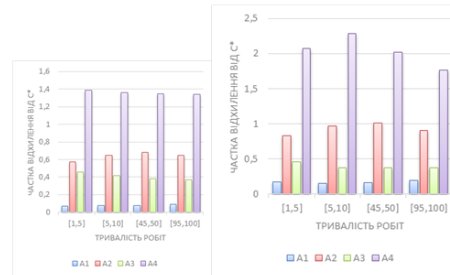
ПЛАКАТ 5 Результати експериментів алгоритмів A1-A4

РЕЗУЛЬТАТИ ЕКСПЕРИМЕНТІВ A1-A4

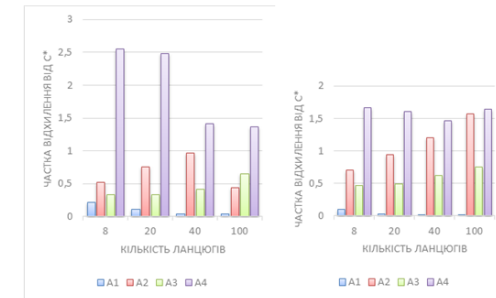
Тривалість робіт (при різних значеннях «ступеня розсіювання» тривалостей робіт)



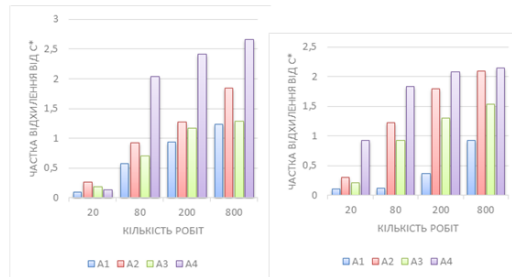
Тривалість робіт (при різних значеннях «ступеня розсіювання» коефіцієнтів продуктивності пристроїв)



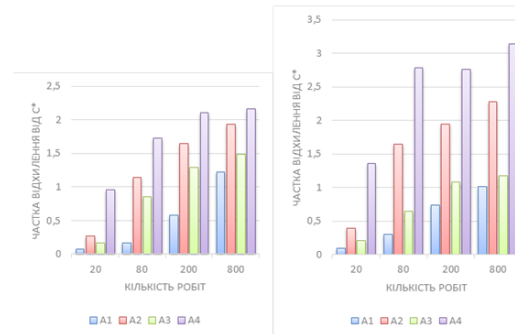
Кількість ланцюгів (при різних значеннях «ступеня розсіювання» тривалостей робіт)



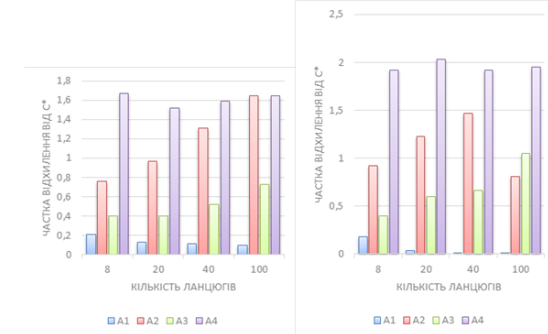
Кількість робіт (при різних значеннях «ступеня розсіювання» коефіцієнтів продуктивності пристроїв)



Кількість робіт (при різних значеннях «ступеня розсіювання» тривалостей робіт)



Кількість ланцюгів (при різних значеннях «ступеня розсіювання» коефіцієнтів продуктивності пристроїв)



Демонстраційний плакат до магістерської дисертації
на тему «Задача складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт»

Виконала студентка гр. ІС-61м

Дубок Катерина Вадентинівна

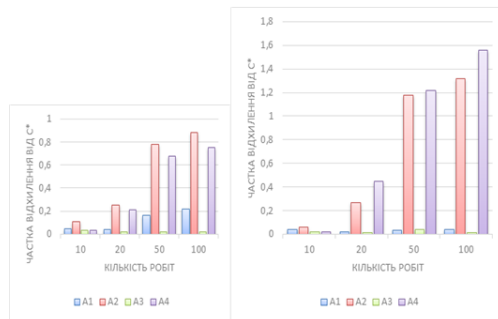
Керівник

Сперкач Майя Олегівна

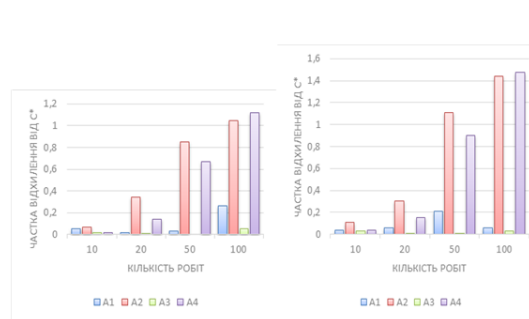
ПЛАКАТ 6 Результати експериментів ПДС алгоритму

РЕЗУЛЬТАТИ ЕКСПЕРИМЕНТІВ ПДС АЛГОРИТМУ

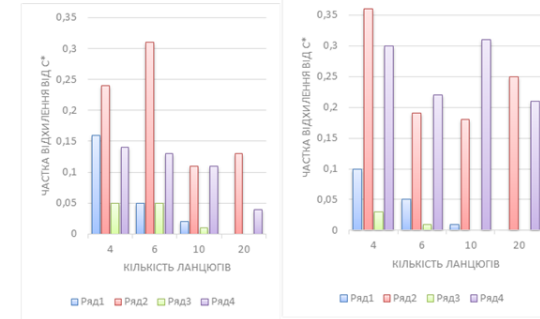
Кількість робіт (при різних значеннях «ступеня розсіювання» тривалостей робіт)



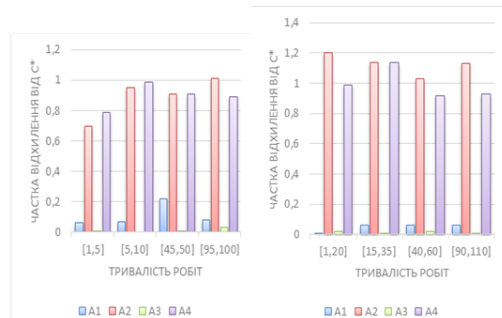
Кількість робіт (при різних значеннях «ступеня розсіювання» коефіцієнтів продуктивності пристроїв)



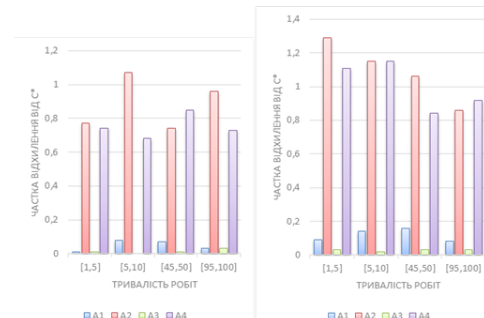
Кількість ланцюгів (при різних значеннях «ступеня розсіювання» тривалостей робіт)



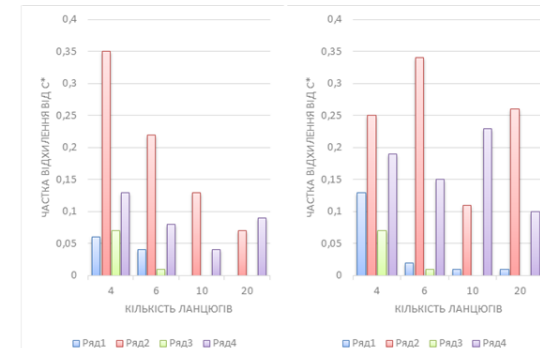
Тривалість робіт (при різних значеннях «ступеня розсіювання» тривалостей робіт)



Тривалість робіт (при різних значеннях «ступеня розсіювання» коефіцієнтів продуктивності пристроїв)



Кількість ланцюгів (при різних значеннях «ступеня розсіювання» коефіцієнтів продуктивності пристроїв)



Демонстраційний плакат до магістерської дисертації
на тему «Задача складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт»

Виконала студентка гр. ІС-61м

Керівник

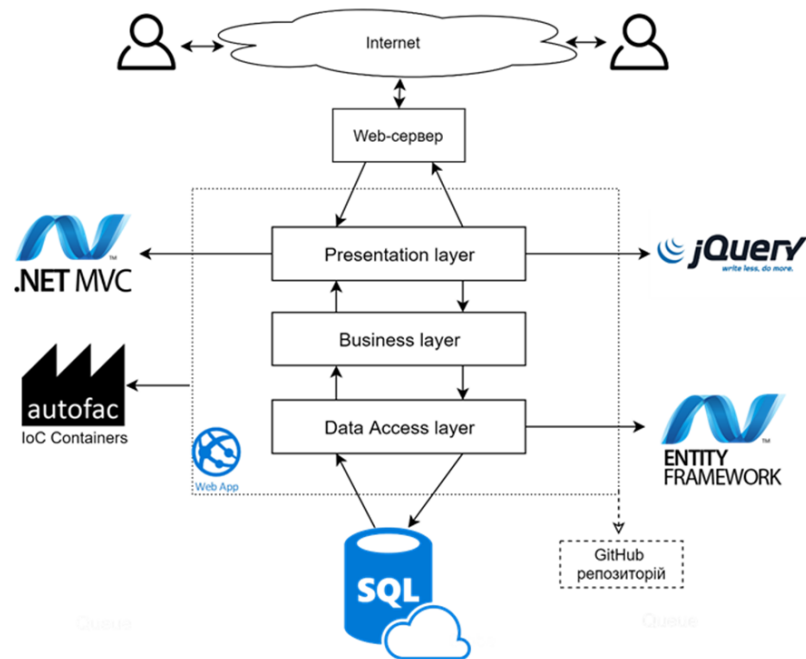
Дубок Катерина Вадентинівна

Сперкач Майя Олегівна

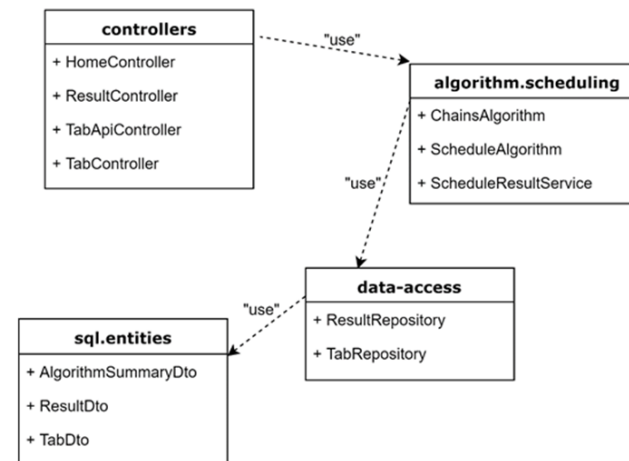
ПЛАКАТ 7 Архітектура програмного забезпечення

АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Архітектура програмного забезпечення



Діаграма пакетів



Демонстраційний плакат до магістерської дисертації
на тему «Задача складання розкладу виконання робіт з відношенням передування паралельними
пристроями за критерієм мінімізації загального часу виконання робіт»

Виконала студентка гр. IC-61м

Керівник

Дубок Катерина Вадентинівна

Сперкач Майя Олегівна

ПЛАКАТ 8 Екранні форми

ЕКРАННІ ФОРМИ

Вкладка «Введення даних»

Складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт

Головна сторінка Введення даних **Наявні дані** Розрахунки

Розрахунок проведено й дані успішно збережені

Пристрої

Кількість: 3

Тип: ☐ Ідентичний ☒ Неідентичний

Пристрій	Продуктивність
1	1
2	1.5
3	2

Палета

Кількість рядів: 2

Кількість об'єктів у ряді: 5

1.5	5
2.1	2
2.2	1
2.3	3
2.4	2
2.5	8

Очистити

Розрахувати

Автозавантаження

Вкладка «Наявні дані»

Складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт

Головна сторінка Введення даних **Наявні дані** Розрахунки

Дані Id:1134 Дані Id:1135 Дані Id:1136 **Дані Id:1137**

Пристрої

Кількість: 3

Тип: Неідентичний

Пристрій	Продуктивність
1	1.0
2	1.5
3	2.0

Палета

Кількість рядів: 2

Кількість об'єктів у ряді: 5

Об'єкт	Час на виконання
1.1	4
1.2	2
1.3	1
1.4	4
1.5	5
2.1	2
2.2	1
2.3	3

Видалити

Результати

Вкладка «Розрахунки»

Складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт

Головна сторінка Введення даних **Наявні дані** Розрахунки

Дані Id:1134 Дані Id:1135 Дані Id:1136 **Дані Id:1137**

Робота	Тривалість
1,1	1.0
1,2	2.0
1,3	4.0
2,1	4.0

Алгоритм	C*	C _{max}
A1	16	16
A2	16	16
A3	16	22
A4	16	16

УМОВНІ ПОЗНАЧЕННЯ

i/j/r- робота на пристрої, де

i - ланцюг

j - порядок виконання роботи в ланцюзі

r - тривалість роботи ij

A1

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 t, од.

C = 12.0 C* = 16.0 C_{max} = 16.0

Демонстраційний плакат до магістерської дисертації
на тему «Задача складання розкладу виконання робіт з відношенням передування паралельними пристроями за критерієм мінімізації загального часу виконання робіт»

Виконала студентка гр. ІС-61м
Керівник

Дубок Катерина Вадентинівна
Сперкач Майя Олегівна